Technische Universität Braunschweig

Development and analysis of a Train-centric Distance Measurement System by means of Colored Petri Nets

Von der Fakultät Maschinenbau
der Technische Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung der Würde

eines Doktor-Ingenieurs (Dr.-Ing.)

genehmigte Dissertation

von:	M.Sc. Haifeng Song
aus:	Henan, China
eingereicht am:	01.03.2018
mündliche Prüfung am:	13.04.2018

Gutachter:	Prof. Dr.-Ing. Dr. h.c. mult. Eckehard Schnieder
	Prof. Dr. Wei Zheng

2018

Acknowledgments

This dissertation is based on my research work in Institute for Traffic Safety and Automation Engineering (iVA) at TU Braunschweig. It would not be possible without the support from many people. I would like to express my great gratitude to all the people who helped and supported me in the last four years.

First, I would like to show my deepest gratitude to my supervisor Prof. Dr.-Ing. Dr. h.c. mult. Eckehard Schnieder. It is honorable that I can be his last Ph.D. student. He gave me not only the research direction but also the attitude toward life. This will be helpful for my whole life.

My gratitude also goes to Prof. Dr. Wei Zheng from Beijing Jiaotong University, who has reviewed this work and provided valuable comments. Many thanks to Prof. Dr.-Ing. Rainer Tutsch from Institute of Production Metrology (iprom) for hosting my doctor defense. He gave me useful suggestions for the defense in a preliminary talk.

I must thanks all the colleagues at iVA for their contribution to an enjoyable working environment. Thanks Susanne Arndt for her language modification of my publications; thanks Dr.-Ing. Uwe Becker, Andreas Dodinoiu, Arne Geffert, Dieter Schnäpp, Geltmar von Buxhoeveden, Jan Welte, Rasmus Rüdiger, and all the other colleagues.

Furthermore, I am very grateful to the Chinese Scholarship Council (CSC) to provide the scholarship for my research. Thanks Mr. Tuo Shen from Tongji University for providing valuable experiment data.

I also would like to thank my girlfriend Aili Li, who helps me both in academic research and daily life.

Last but not least, I wish to thank my parents Shude Song and Yanzhen Wang, who decided 29 years ago to give birth to me. Even though they have no idea about my work, they encourage and stand by me all the time.

Braunschweig, June 2018 Haifeng Song

Contents

List of Figures

List of Tables

Kurzfassung

Basierend auf technologischen Trends sollte das Zugbeeinflussungssystem den Anteil der Bodenanlagen reduzieren und den Zügen mehr Eigeninitiative geben als in der Vergangenheit, da so die funktionale Sicherheit und die Flexibilität des Zugbeeinflussungssystems erhöht werden können. In dieser Arbeit wird ein verbessertes System vorgeschlagen, das die Vorteile der zugbezogenen Kommunikation mit den aktuellen Fahrbefehlsmechanismen kombiniert. Um die notwendigen Daten des Zugabstandsintervalls zu erhalten, werden die Bordausrüstung und ein neues Zug-zu-Zug-Entfernungsmesssystem (TTDMS) als normale bzw. Backup-Strategien angewendet.

Während verschiedene Ortungstechnolgien zur Zugdatenerfassung genutzt wurden, bleibt die Entwicklung und Validierung neuer Systeme eine Herausforderung. In dieser Arbeit werden formale Ansätze zur Entwicklung und Verifikation von TTDMS vorgestellt. Zur Unterstützung der Systementwicklung werden CPNs zur Formalisierung und Bewertung der Systemstruktur und ihres Verhaltens eingesetzt. Basierend auf dem CPN-Modell wird die Systemstruktur validiert. Zusätzlich wird eine Methode vorgeschlagen, mit der eine Code-Architektur aus dem formalen Modell generiert werden kann. Die Systemleistung wird im Erfassungsbereich und in der Genauigkeit beurteilt. Daher werden sowohl eine mathematische Simulation als auch eine praktische Validierung der Messungen implementiert. Die Ergebnisse zeigen, dass das System in der Lage ist, Entfernungsmessungen in Metro- und Eisenbahnlinien durchzuführen. Zudem sind die formalen Ansätze bei der Entwicklung und Verifikation anderer Systeme wiederverwendbar.

Die Abstandsmessung mit TTDMS basiert auf einem Frequenzspreizungsverfahren. Die Messung wird durchgeführt, indem die Ankunftszeit angewendet wird, um den Abstand zwischen zwei Zügen zu berechnen. Dieses Verfahren erfordert keine Synchronisierung der Zeitquellen der Übertragung. Der Zeitunterschied kann damit berechnet werden, indem die Autokorrelation des Pseudo-Random-Noise-Codes verwendet wird. Im Unterschied zu Systemen im Luft- und Seeverkehr benötigt dieses System keine andere Lokalisierungseinheit als die Kommunikationsarchitektur.

Um zu gewährleisten, dass ein System wie vorgesehen funktioniert, muss es validiert werden. Nur wenn das Systemverhalten validiert wurde, sind Bewertungen anderer relativer Leistungen sinnvoll. Aufgrund ihrer eindeutigen Definition kann das TTDMS mit formalen Methoden viel klarer beschrieben werden als mit ausführbaren Codes.

Abstract

Based on the technology trends, the train control system should weaken the proportion of ground facilities, and give trains more individual initiative than in the past. As a result, the safety and flexibility of the train control system can be further improved. In this thesis, an enhanced movement authority system is proposed, which combines advantages of the train-centric communication with current movement authority mechanisms. To obtain the necessary train distance interval data, the onboard equipment and a new train-to-train distance measurement system (TTDMS) are applied as normal and backup strategies, respectively.

While different location technologies have been used to collect data for trains, the development and validation of new systems remain challenges. In this thesis, formal approaches are presented for developing and verifying TTDMS. To assist the system development, the Colored Petri nets (CPNs) are used to formalize and evaluate the system structure and its behavior. Based on the CPN model, the system structure is validated. Additionally, a procedure is proposed to generate a Code Architecture from the formal model. The system performance is assessed in detection range and accuracy. Therefore both mathematical simulation and practical measurements validation are implemented. The results indicate that the system is feasible to carry out distance measurements both in metropolitan and railway lines, and the formal approaches are reusable to develop and verify other systems.

As the target object, TTDMS is based on a spread-spectrum technology to accomplish distance measurement. The measurement is carried out by applying Time of Arrival (TOA) to calculate the distance between two trains, and requires no synchronized time source of transmission. It can calculate the time difference by using the autocorrelation of Pseudo Random Noise (PRN) code. Different from existing systems in air and maritime transport, this system does not require any other localization unit, except for communication architecture.

To guarantee a system can operate as designed, it needs to be validated before its application. Only when system behaviors have been validated other relative performances' evaluations make sense. Based on the unambiguous definition of formal methods, TTDMS can be described much clearer by using formal methods instead of executable codes.

1 Introduction

In this chapter, first, the purpose of this thesis is discussed, which applies Colored Petri Nets (CPNs) to assist the development and analysis of a train to train distance measurement system (TTDMS). This system, which together combines the train-centric communication technology, intends to enhance the movement authority and improve the train operation safety. Second, the structure of thesis is introduced.

1.1 Purpose of the Dissertation

The development of system leads to a progressively complex system structure. The well-structured and concise descriptions of procedures should be involved in the assistance of the system's development. An efficient strategy to solve this issue is to apply formal methods. The higher the formalization level used to describe the system, the greater the possibility to formally verify the formalized concept of the system. Hence, high-level Petri nets, CPNs are suitable for the development and representation process. The system structure is formalized in a CPN model. After confirming the safety of the model by means of model based validation, the model can provide a framework for the programming of the code structure.

The overall purpose of this thesis is to assist the development of a new railway signaling assistant system, and to analyze the safety and availability of it using Petri nets. This new system development starts from the requirements analysis, and then it will be involved in existing systems as an overlay system, the overlay system's safe and availability is also considered. The model is validated with a case study of a train-centric communication signaling system with different scenarios. This purpose can be decomposed into five objectives, as shown in Fig. 1.1.

The first objective is analyzing the system requirements and feasibility. By investigating the weakness of the current system, the improvement direction can be obvious. Before the new system is applied, evaluation of the safety improvement is the next task. In particular, this objective tries to answer: what kinds of problems can be solved?

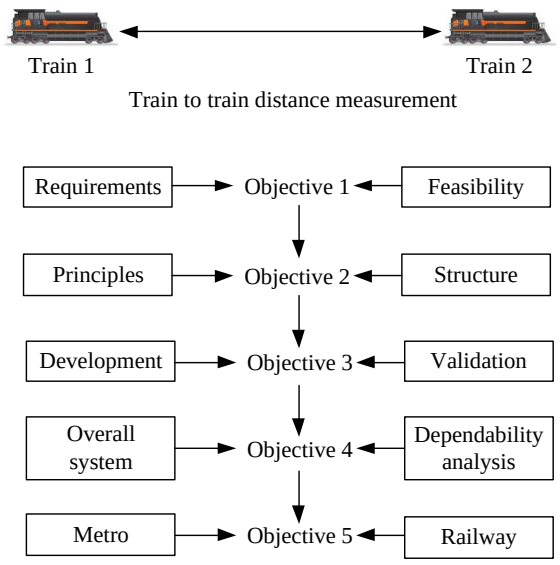


Figure 1.1: The relations of the objectives and thesis

The second objective is to evaluate the system performance. To improve the system function, different kinds of technologies are available to be applied. This objective is aiming to answer the questions: Which kind of technology to be chosen to carry out the system function, and how to design the system structure? What is more, how the efficiency of this system is estimated?

The third objective is the advancing of the system development and analysis procedure. Since CPN method is chosen as the formal analysis methodology, this objective is going to answer the questions: how to integrate the system development and formal method together? First, the second objective has to be reflected in a CPN model, which can be used to carry out the validation and verification. Second, with proper parameters, the system performance can be evaluated by using simulation.

The fourth objective is the assessment of the overall system performance. When the system is developed, we have to evaluate the overall system performance. Time-related system states can be used to evaluate the dependability of complex systems which are widely analyzed with the fault tree. As the Fault Tree is limited to many commercial products, the CPNs are applied to represent and evaluate the fault tree, and to estimate the overall system performance.

The last objective is the practical implementation of the system. After the aforementioned four objectives are achieved, this objective shows the actual industrial applications. The

system has been applied in metro lines, and the feasibility of implementing in railway systems is analyzed.

1.2 Structure of the Dissertation

This thesis consists of 11 chapters, both methodologies and numerical analysis results are provided. The whole structure of the thesis is shown in Fig. 1.2. The first two chapters contain previous experience and preliminaries, which involve the state of the art and relevant basic knowledge.

Chapter 1 describes the purpose and structure of this thesis.

Chapter 2 investigates the state of the art and challenges of different sorts of train control systems. As the most advanced and widely applied train control system, the European Train Control System level 2 (ETCS-2) preliminaries are introduced. After that, the distance measurement methods in railway are presented. With this background, the train collision accidents and the methods to prevent them are described. Additionally, the train-centric communication control system and its associated challenges are presented.

Chapters 3 to 10 are my original personal work. These chapters:

- propose a train-centric based train distance measurement system, and analyze the performance improvement by involving this system;
- establish a procedure for the development, validation and verification of TTDMS by means of CPN models;
- represent and evaluate the fault tree in CPN models to assess the overlay system performance;
- discuss the system's practical implementations in metro and railway, analyze both the simulation and actual measurement data.

Chapter 3 discusses the necessity of involving the new system to improve the overall railway system safety. The overall system safety improvement is calculated based on the system structure. For detailed analysis, a train to train collision failure model, which will be evaluated by means of CPNs in the chapter 8, is introduced.

Chapter 4 illustrates the enhanced movement authority system (MA+), which is a train-centric based system. The system principle, structure, and algorithms are introduced. MA+ has two strategies to obtain the train distance interval data. The normal strategy is based on

the existing onboard data to carry out the distance calculation. While, the backup strategy is implemented by TTDMS, which uses the Time of Arrival (TOA) method to implement the distance estimation. What is more, the reliability of an actual prototype machine is estimated. With this system involving and combining it with the ETCS-2, the train safe distance is shorter than it in the ETCS-2 and ETCS-3.

Chapter 5 is going into detail about the development and analysis of TTDMS by using CPNs. First, evaluation methodologies are discussed. Second, in order to assist the development and analysis of a new system, the CPN based methods are involved. The basic definitions of CPNs, which are chosen as the means of the description of the system design in this thesis, are introduced. Besides, different Petri net tools are compared, and the *CPN Tools* and $\pi-$Tool are applied in the following chapters. The basic instructions, such as referring to the items in a state space and data value, are introduced briefly. Finally, the task is divided into four steps and four different levels, and further discussed in the following chapters.

Chapter 6 presents the formal modeling, validation and verification process of the TTDMS. The system property concept is used to build the CPN model. Model validation and functional safety verification are implemented through the model checking method. For the performance evaluation, the parameterization process is introduced, and the simulation data is obtained by Monte-Carlo-Algorithm testing.

After the system structure and performance is validated and evaluated at the model level, chapter 7 proposes the procedure for the code framework generation and model based execution time estimation.

Chapter 8 depicts the improvement of the new system in the train collision prevention. Because fault tree analysis is limited to assess dynamic systems without event-repair operations and probability-related attributions, its analysis and evaluation can not satisfy actual requirements. Hence, in this chapter, the CPNs are applied to represent and evaluate the fault tree. With the methodology introduced in this chapter, the collision model introduced in chapter 3 is analyzed in the CPN model.

Chapter 9 illustrates the implementation of TTDMS in operational metro lines. A speed based distance warning strategy is proposed. The system availability is evaluated by the stochastic Petri nets. The detection distance is validated in both simulation model and actual scenarios.

Similarly, the proposal of applying MA+ in the high-speed railway is illustrated in Chapter 10. As limited by the hardware and experiment requirements, only simulation results are

available for the reference. Additionally, an application demo on the DMI is proposed, even though further validation and analysis are required.

With the methodologies and numerical results, chapter 11 concludes the thesis and indicates the further work. The system proposed in this thesis has been applied in actual metro lines, which can be referred for the further development.

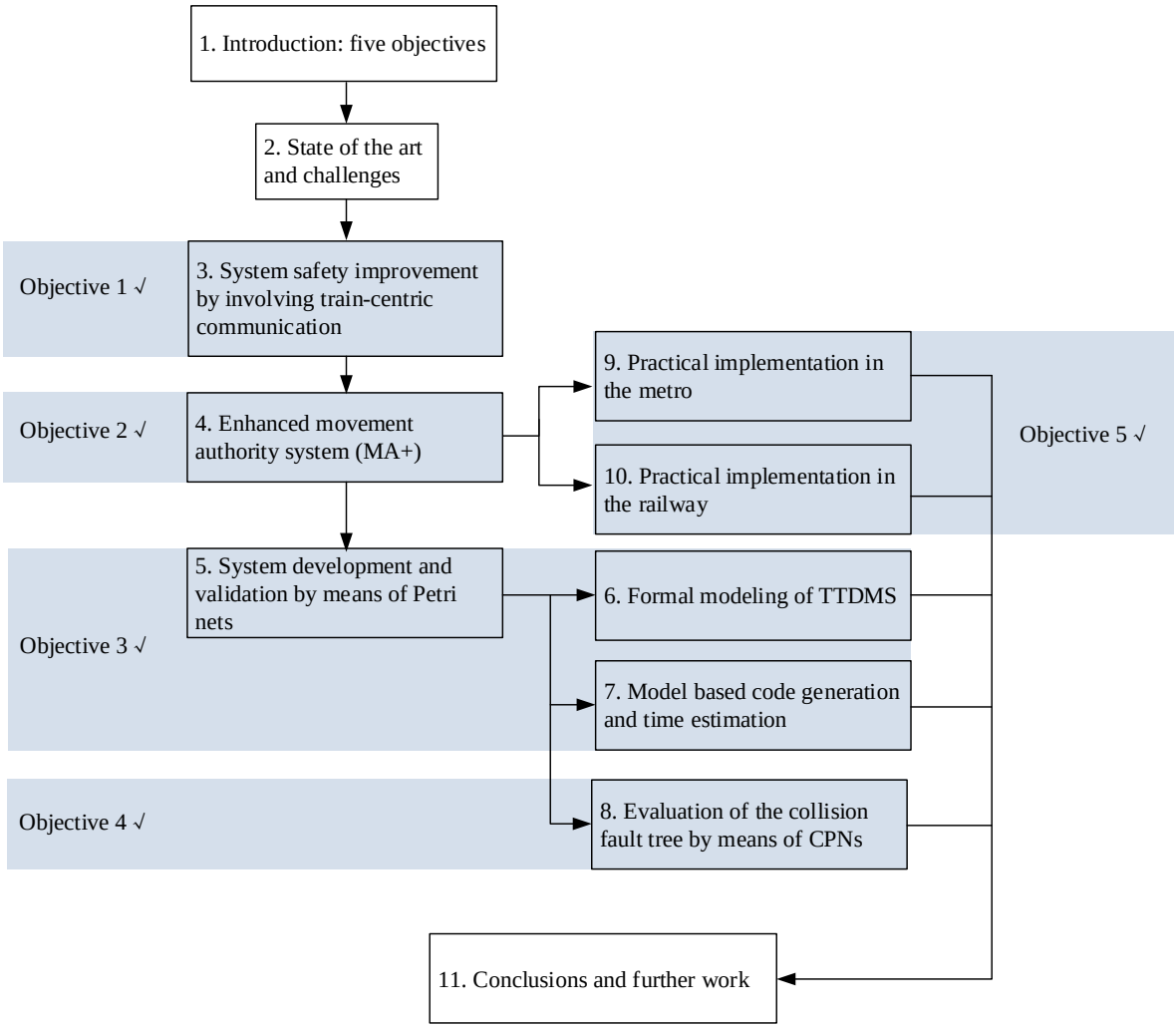


Figure 1.2: Structure of the dissertation

2 State of the art and challenges

In this chapter, first, the different train control systems and the preliminaries of ETCS-2 are discussed. Second, the distance measurement methods in railway are proposed. Third, the train collision accidents, as well as the avoidance system are presented. Finally, as the emerging technology, the train centric communication control system and associated challenges are introduced.

2.1 Train control system

Train control systems have gone through a rapid evaluation since the first high-speed train started operation in Japan in 1964 [2]. Take the European Train Control System (ETCS) as a case study, which is specified at four levels:

- Level 0 requires train driver to observe the trackside signals, and the maximum speed is monitored by the onboard equipment.
- Level 1 is a cab signaling system, which can be superimposed on the existing signaling system. It involves track-side signals, balises, and track circuits. Radio information and balises transmit the signal to the vehicle as a movement authority together with route data at fixed points.
- Level 2 is a digital radio-based system. The Movement Authority (MA) and other signal aspects are displayed in the cab for the driver. It involves balises, track circuit, and GSM-R.
- Level 3 is currently under development. It relies on the onboard equipment to carry out the train location detection and integrity supervision, but no the trackside equipment such as track circuits or axle counters in level 2.

Among these four different solutions, the ETCS level 2 (ETCS-2) is widely applied in nowadays especially in high-speed lines. This section seeks to show some preliminary notions that

are necessary for the discussion that will be conducted in the following chapters. We assume that the reader has some basic knowledge of ETCS, otherwise, ERTMS/ETCS system requirements specification [3] are highly recommended for starters.

The basic components of ECTS-2 are shown in Fig. 2.1. The train is moving based on the MA, which is transfered from the Radio Block Centre (RBC) to the train. The MA in ETCS can be depicted in Fig. 2.2. From the current train position to the End of Authority (EOA), the distance is composed of several sections [3]. The MA details can be obtained from the Packet 15 in wireless Message 3. Table 2.1 indicates the practical definition of Packet 15, which contains the details of Level 2/3 Movement Authority. Message 3 is the movement authority in the radio message, and its standard definition as shown in Table 2.2.

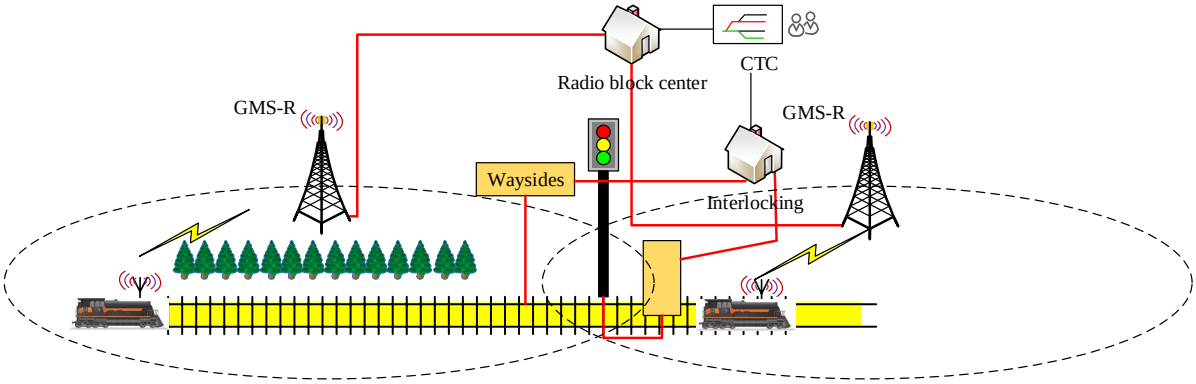


Figure 2.1: The basic components of ETCS-2

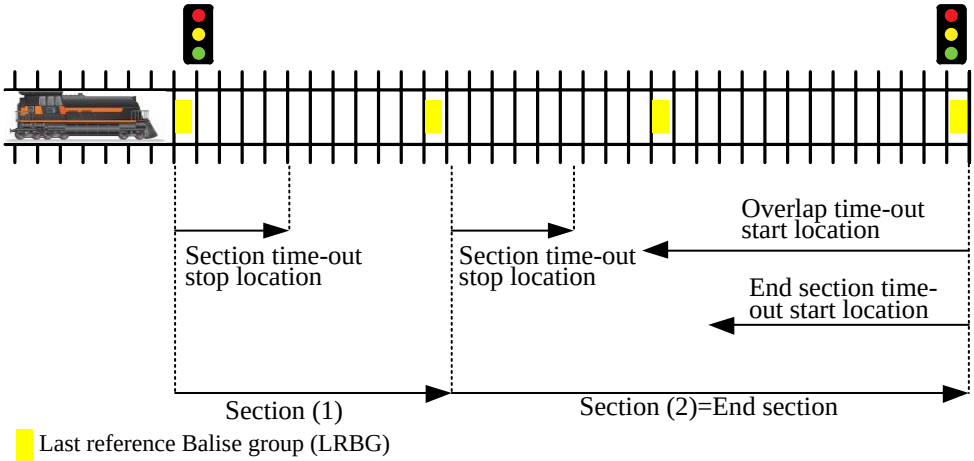


Figure 2.2: Structure of an MA in ETCS

For each section composing MA the following information shall be given [3]:

Table 2.1: Packet 15 (partial)

Variable	Comment	Value
NID_PACKET	Packet ID	15
Q_DIR	Validation direction	0/1
L_PACKET	Packet length	depends on content
Q_SCALE	Distance length	1(1 m)
V_LOA	LOA speed	0
L_SECTION(k)	Section(k) length	depends on operation details
L_ENDSECTION	End section length	depends on operation details

Table 2.2: Message 3: movement authority

Field No.	Variable	Comment	Value
1	NID_MESSAGE	Message ID	3
2	L_MESSAGE	Message length	depends on the message content
3	T_TRAIN	On board equipment time	depends on the message sending time
4	M_ACK	Feedback confirmation	1
5	NID_LRBGE	Last Reference Balise Group	the last NID_LRBG in the train location report
6	Level 2/3 MA	Movement authority	Packet 15
7	Optional packets	Optional	Relevant packets (15, 5, 21, 27, 3, 41, 65, 68, 80)

- length of the section, presented in Packet 15 as *L_SECTION(k)* and *L_ENDSECTION*;
- optionally, section time-out value and distance from the beginning of Section Time-out stop location.

Hence, the MA length can be calculated with the variables of *L_ENDSECTION* and *L_SECTION(k)*, as shown in equation (2.1).

$$L_{MA} = \sum_{k=1}^{n} Section\,(k) + EndSection \tag{2.1}$$

where $Section(k)$ and $EndSection$ denote the length of the End section and the section k in the MA, respectively [3].

After obtaining the MA, the onboard equipment displays the essential information through the Driver Machine Interface (DMI), as shown in Fig. 2.3. The DMI carries out two main functions to:

- display information to the driver in response to operational situations. There is visual

information, such as symbols and text messages, as well as audible information for the speed and distance monitoring;

- provide the interactions between the driver and onboard equipment. These include data input and acknowledgments by the driver.

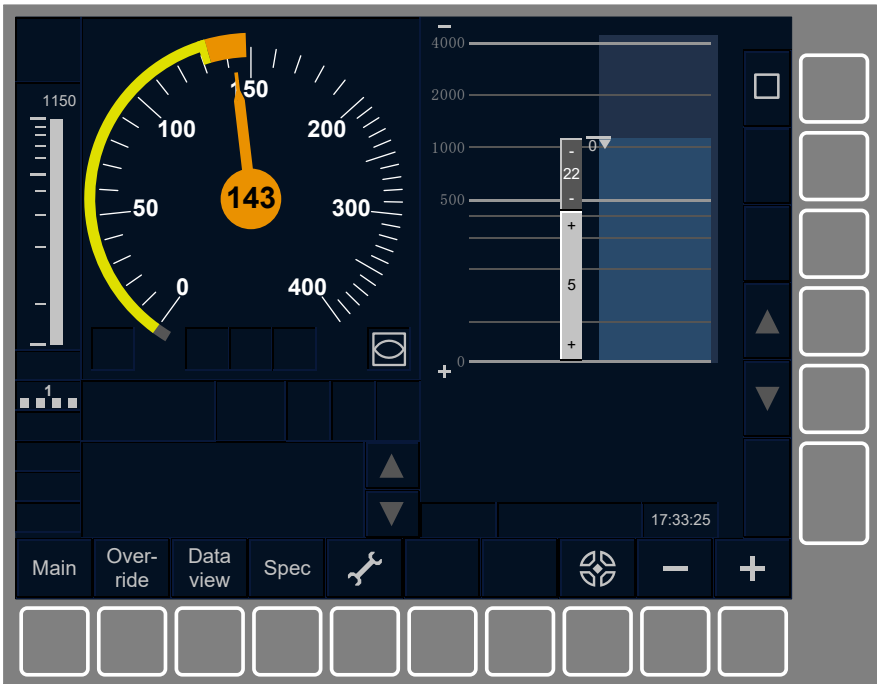


Figure 2.3: ETCS Driver Machine Interface

On the DMI, the orders and announcements overview shall be displayed within the movement authority and up to the first target at zero speed, the following aspects should be involved if any, as shown in Fig. 2.4.

- Distance scale
- Orders and announcements of track conditions (excluding tunnel stopping areas)
- Gradient profile
- Speed profile discontinuities
- Planning Area Speed Profile (PASP)
- Indication marker
- Hide and show planning information
- Zoom function

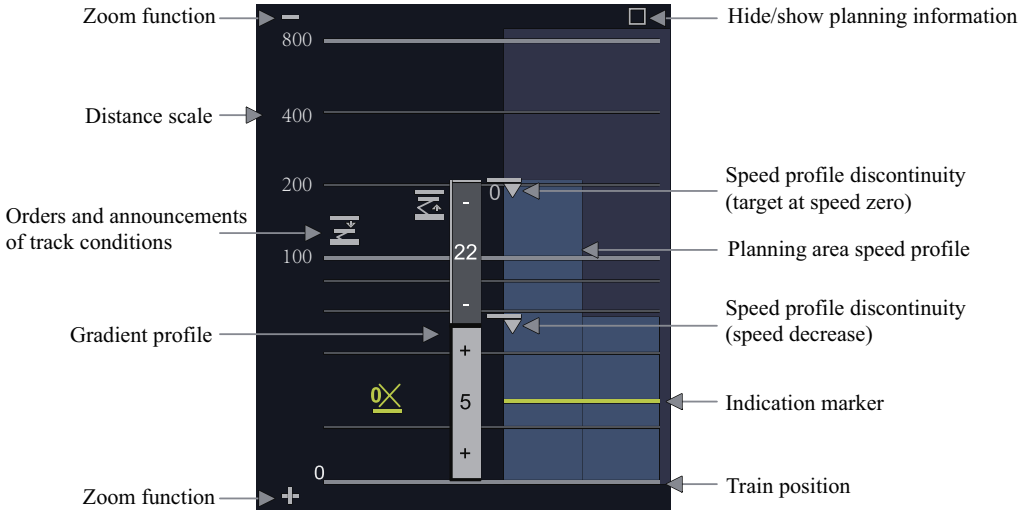


Figure 2.4: Main objects of the planning information

2.2 Distance measurement methods in railway

Localization is one of the basic tasks of a train control system. Train control needs the position information to control and manage the transportation. Position can be divided into absolute position and relative position. Nowadays train control systems are using the absolute position to control and manage trains operation.

Physical space separation is a fundamental safety precaution for trains. The primary goal is to obtain the distance information between two trains in time. The means of acquiring this distance information is to collect the location data of the train. Different technologies can be applied in this process, as shown in Fig. 2.5. Various methods, systems, and tools can be used to track the train's absolute position. For instance, track circuit, axle counter, Global Navigation Satellite System (GNSS), Balise, Radio frequency identification (RFID), Wireless Local Area Network (WLAN), and Ultrasonic range finding [4]. Alternatively, radar, speed sensor, and spread spectrum communication can obtain the train's relative position data.

The primary solutions are track circuit and axle counter. This kind of method has a relatively high reliability, while the location resolution accuracy is low. Balises in ETCS-2 refer the absolute position. Its absolute localization accuracy is affected by the layout density of balises. Balises can also provide onboard systems with additional data [5]. These devices store static or dynamic information depending on their types. However, Balises constitute a discontinuous communication system. What is more, balise is expensive and maintenance is required. There is also other hardware (accelerometer, gyroscope, odometer, etc.) which can generate relative position information, however, the disadvantage is as time goes on,

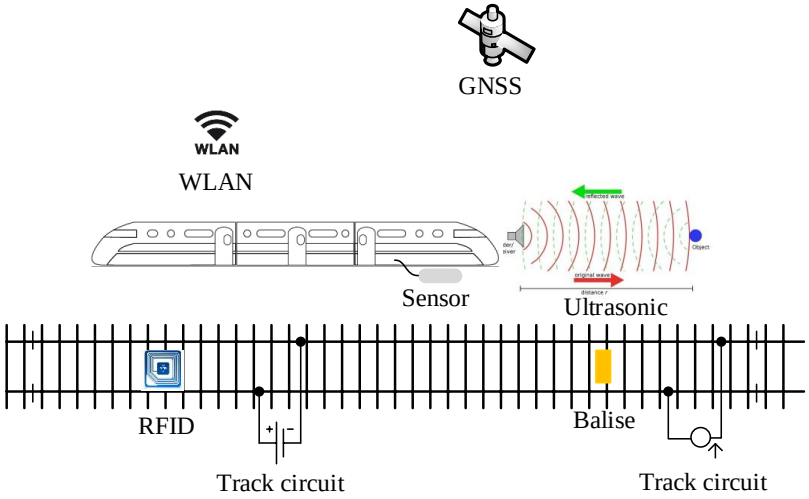


Figure 2.5: Localization methods

accumulative errors will have a certain increase.

GNSS is also one of the most developed location technologies, and it has been applied in different research areas [6–9]. The advantages of GNSS are high location precision and low costs. However, GNSS can only operate in an open area to receive location information [10]. However, due to GPS inherent error (multipath, ionospheric, mountain block, and so on), GNSS solely based systems have not enough accuracy. Additionally, GNSS cannot provide satellite signals for metro trains that are operating inside the tunnels.

WLAN location technology requires additional database support, because fingerprint-labeled radio maps are needed. Ultrasonic location technology uses the Time of Arrival (TOA) to calculate distance, but the location results are not accurate inside the tunnels because of the severe beam divergence. RFID enables a one-way wireless communication. However, the RFID tag must be put on objects previously [11].

Wireless location technology combines both location and communication together, and it can be used to calculate relative distance between two trains. The procedure is about the communication and distance calculation between two different Mobile Stations (MS), not like nowadays wireless location system, which can obtain absolute distance and position in a Cell Coverage Area. Fig. 2.6 shows a conventional location system.

Wireless location methods can be divided into three main parts, which are: signal intensity, time differences of arrival (TDOA), and Angle of Arrival (AOA), respectively. Signal intensity relied on the accuracy of path loss model, which will be variable at a different distance and affected by some other factors (wall, metal, vehicle, and so on). Hence, the location accuracy is relatively low. AOA needs antenna array to realize location, which has a high

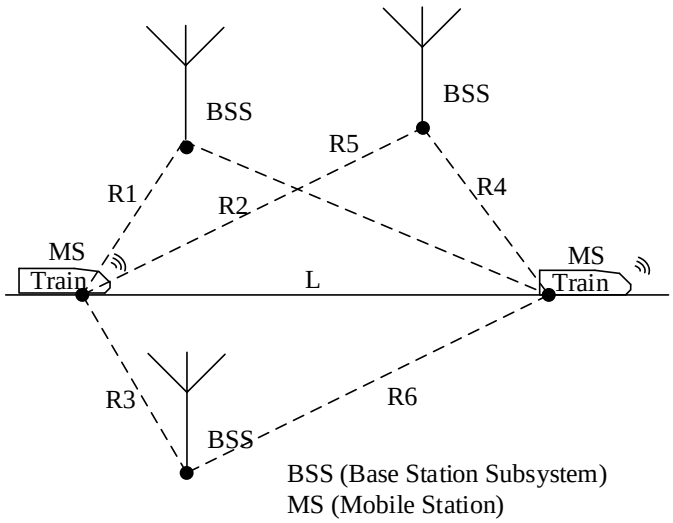


Figure 2.6: Wireless location technology

system requirement in the MS part. What's more, in long distance measurement, a tiny deviation in the angle can lead to a great difference in distance measurement result.

Using MS to calculate the distance between two trains, some deviations cause by normal wireless locations system can be solved. For instance, multipath fading, near-far effect, and non-line-of-sight propagation (NLOS). In the point-to-point distance measurement system, only two MS are involved in the distance measurement. Consequently, the near-far effect is not adaptive in this scenario. For multi-path fading, already many successful methods can be applied to solve this problem, for instance, Root-Music and ESPRIT method [12, 13].

Contrary to NLOS propagation, the corresponding one is line-of-sight (LOS) propagation. Under the typical railway situation, LOS propagation in point-to-point communication will cause distance measurement deviation in curve lines, as shown in Fig. 2.7. When the train is running in the curve line, the LOS distance is shorter than the curve line. The real distance between two trains in the track is $\widehat{APB}$. However, the calculated distance is ALB, which will be shorter than the actual distance. The error caused by LOS curve lines can be found out in following Fig. 2.7.

2.3 Train collision accidents and avoidance

According to International Union of Railways (UIC) and European Railway Agency (ERA) definition, types of railway accidents can be divided into three aspects: collective, individual,

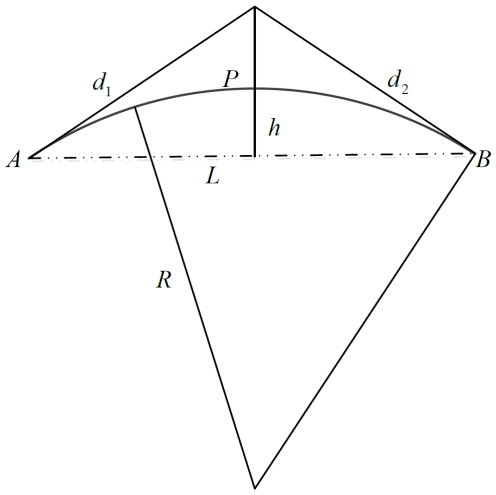


Figure 2.7: Error caused by LOS

and other types of accidents. Actual statistics of the UIC show that there are still significant train accidents in Europe every day. Among these accidents, collective accidents cause more severe economic and personal losses than other two accident types. The collective accidents involve derailment of trains, train collision with another train, and train collision with an obstacle. Fig. 2.8 indicates the comparable data of UIC statistical reports, the number of accidents increased markedly in Europe[1]. Even though the latest technologies have been implemented, accidents happened all the same. Hence, the process to enhance railway safety can never be stopped.

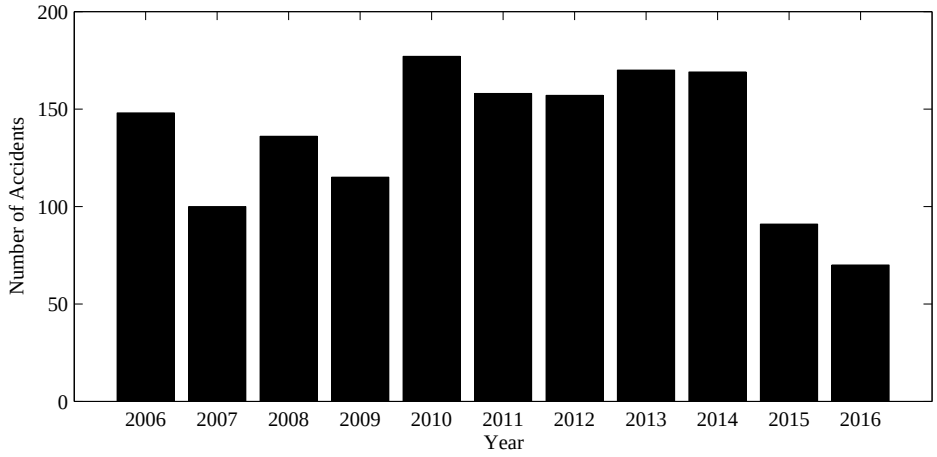


Figure 2.8: Train collisions and derailments numbers of events in European, 2006-2015, International Union of Railways

[1]Data resources: UIC Safety Database Activity Report 2007-2016

These three significant train accidents happen in Europe every day, despite millions of Euros which have been invested in trackside and in-train safety equipment. Even with Automatic Train Protection (ATP) systems like the future European Train Control System (ETCS) a significant amount of accidents cannot be prevented. Since the accidents occur between trains and other kinds of obstacles like construction vehicles, construction workers or pedestrians and vehicles on level crossings. Additionally, the control system could also be out of service, and the human error is also the main reason.

If the signaling equipment do not guarantee the fail-safety, the train distance interval would no longer be monitored, which may result in train collision accidents [14, 15]. For instance, in 2016 Berlin Tram-Crash, two commuter trains collided on a single-track of railway in Bavaria, Germany [16]; a head-on collision involving two passenger trains in southern Italy as well as; two cargo trains collided in Finland is the further example; the 7.23 Yong-wen line train collision [17], the Shanghai Metro Line 10 collision in China [18]; collision between a Union Pacific freight train and a METROLINK Commuter train in the USA [19], and so on.

In today's Train Control System, there are a number of technologies, processes, and procedures that are combined to ensure the safe operation for trains. They are working together to ensure that the risk of train collision is consistent with an acceptable Target Level of Safety (TLOS): *The target level of safety is the probability of an accident (fatal or hull loss) during a movement.*

These capabilities effectively function in a layered approach. Similar to defense in-depth and layered information safety architectures, it would take failures at multiple layers to cause a system failure, resulting in a collision. The risk of collision can be described as a layered approach, as shown in Fig. 2.9. It takes failures at multiple layers to cause a system failure [20]. Layer one indicates structures and procedures of the train operation. This layer is defined in a manner that further reduces risk from the pure random chance, and does not specifically focus on mitigating the risk of a collision between two trains. The timeliness of layer one is lower when compared with other four layers. As an instance of the layer one, the train diagram separates trains in timetables and reduces the potential conflicts, but it has to be modified based on the actual operation of the train transportation. As the two and three layers, blocks and movement authority implemented by advanced signaling systems indicate where the train is allowed to reach, and increase the train transport safety. In layer four, the railway collision avoidance system services as an assistant system, which effectively reduces the risk of collision [21]. The last layer is "Driver see and avoid", the driver directly controls the train and can react to the accident.

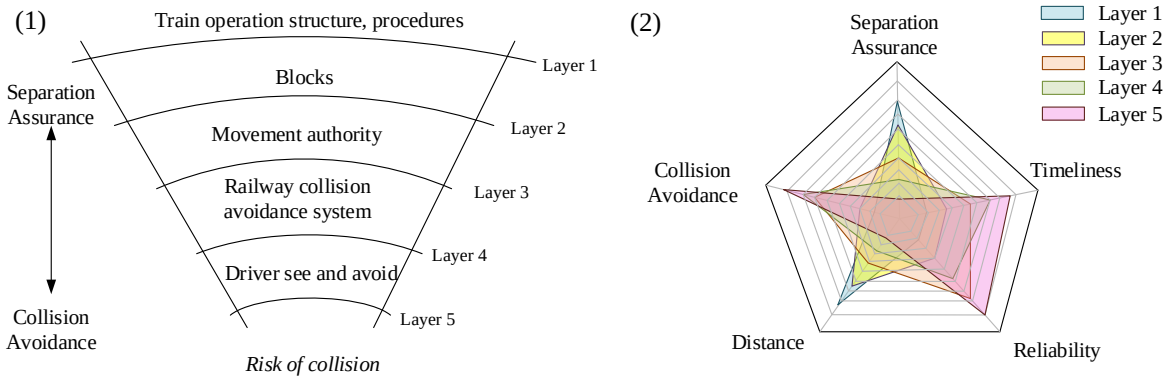


Figure 2.9: Risk layer of collision

Contribution [14] discusses failure events of the train derailments and collisions, and this reference is highly recommended for beginners. Once a hazardous movement is accepted by the train, the driver does not know the train is operating a dangerous action until accidents like collisions or derailments became apparent.

2.4 Emerging technologies: Train-centric communication control system

To carry out the function of these five layers and to ensure the train safe distance interval, the train control system plays a very significant role.

In order to meet the increased safety requirements in railway traffic, aforementioned train control systems are installed in some countries and areas. Most of these control systems are centralized, especially the Automatic Train Protection (ATP) system. Here the trains are monitored by devices located along the rail. These devices send the collected information to an operation control center that can send specific instructions to the train. A European ATP standard, European Train Control System (ETCS), is intended to replace the various European ATP systems, in order to protect international train traffic. However, according to the estimations of the German railway company "Deutsche Bahn" (DB), it could take up to 20 years and cost up to 8 billion Euros to introduce ETCS right across Europe. Additionally, only the operation control centers have an overall overview of the traffic situation, and a train driver could only be warned of a hypothetical collision if the operation control center decides it.

Hence, some projects are carrying out the research of train-centric communication control system, which provides a direct communication channel for trains. As a result, it enables "virtually coupled trains" to operate much closer to one another (within their absolute braking distance) and dynamically modify their own composition on the move (virtual coupling/uncoupling of train convoys), while ensuring at least the same level of safety as is currently provided. For example, the Shift2Rail joint venture project contributes to:

- Cutting the life-cycle cost of railway transports by as much as 50%
- Doubling railway capacity
- Increasing reliability and punctuality by as much as 50%

In this project, wireless virtual coupling of trains is further developed. With the assistance of train to train communications, the virtual coupling breakthrough concept will revolutionize the paradigm of railways [22]. Similar research is done in the Next Generation Train (NGT) project [23]. The coupling will be non-mechanical, but virtual and will be maintained by means of distance control technology. A communication link between the trains can ensure that if the leading train starts to brake, the following train will do the same and maintain separation as the two trains slow together.

Based on the technology trend, the train control system should reduce the proportion of ground facilities [24]. Different from the centralized management systems, some projects based on the train-centric communication have been carried out in these years.

For instance, Alstom offers a new train-centric communication-based train control system, which called Urbalis Fluence. Urbalis Fluence is a CBTC solution whose intelligence is based onboard. This system can carry out the direct train-to-train communication, and the trains can control switches independently. The train-centric architecture brings better performances with less equipment. The train-to-train communication significantly reduced response time for lower headway. What is more, the track resources directly booked by the train, and the simple architecture can improve system availability and reduce maintenance cost. As a result, this system's maintenance costs have been decreased by 20%, and energy has been saved up to 30% [25].

In order to increase safety in railway traffic, German Aerospace Center introduces a train collision avoidance system (RCAS), which is a vehicle integrated collision avoidance system similar to the existing ones in maritime or air transportation. The basic idea of RCAS is to calculate the own position and movement vector and broadcast this information as well as additional data like vehicle dimensions to all other trains in the area. This system permits

the train-to-train communication, and potential collisions are avoided in the future based on the system [26]. RCAS can be applied in the scenarios like: regional lines, train stations, and shunting yards. Main lines with high-speed services are not considered, because there the safety level is already very high due to extensive technical equipment and train protection and control mechanisms.

2.5 Challenges associated with the Train-centric communication system

The challenges associated with the train-centric communication system involve two aspects: development and implementation.

2.5.1 The system development

The system development procedure involves five different phases: planning, analysis, design, implementation, and maintenance. These five phases involve different staffs and knowledge. Hence, how to connect each phase and make sure an efficient information communication is becoming a challenge. Additionally, the development of safety-critical and also multipurpose automation systems leads to more and more complex control structure, especially for discrete and hybrid systems. One approach to solve the problem and to improve the results is to use so-called formal approaches [27].

For the safety analysis of the system, the formal approaches have been widely used in the analysis and verification of the train control signal system. For instance, the formal methods based on CPNs, Unified Modeling Language (UML), automata theory, and so on [28–30].

However, the aforementioned methodologies generally suffer from the following shortcomings:

- Most studies are merely analyzing the system's partial requirements and characteristics, and lack of effective methods to model and verify the entire system development and evaluation;
- In some studies' (such as literatures [31–33]), the system's modeling process and actual development are independent. There is a lack to combine these two procedures together.

- The heterogeneity between the safety model and the system development makes it difficult for the model conversion code to be reused.

Given the above issues, this thesis will carry out an in-depth research to promote the signal-related system modeling, verification and development work further development; to establish an assistant procedure that starts from the concept of the system to the system design and system implementation [34].

Different formal methods are available to be applied, and we should find one that can cover as more requirements as possible. In this thesis, the Petri net is selected. The benefits and reasons are introduced in chapter 5.2. The relationship between Petri net and the actual system behavior has to be deeply explored, and the system logical structure needs to be accurately expressed by Petri net model. The functional safety of the system is verified by the model test method of CPNs. The Monte Carlo simulation experiment is carried out by parameterizing the model to obtain the statistical analysis data of the system. Moreover, based on the safe model produced by the state space, a safe code framework will be generated. Finally, the code framework will be applied to the actual system programming to save the time spent on system development to ensure that the safety function calls between the code.

2.5.2 The system implementation

The train-centric communication system carries out the data exchange between trains. The fundamental tasks are: obtaining the data and transferring it, which are related to the sensor technology and communication frequency selection, respectively.

Sensor Technology

Sensor inputs vary in terms of their data content and coordinate systems, their fields of view, accuracies, and update rates. The task of sensing obstacles in the environment can be performed by different kinds of sensors, most of which are limited to one degree of freedom. The fundamental information that a sensor needs to acquire is the range, azimuth, and elevation of all targets of interest, as shown in Fig. 2.10. If a suite of diverse sensors is used to detect different targets, the algorithms must deal with these data differences. From the view of sensors, two different ways to obtain data: active and passive.

There are two fundamental types of sensors to perform the surveillance of collision avoidance:

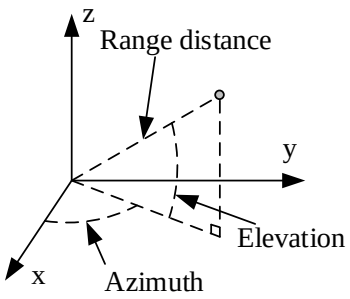


Figure 2.10: Sensor degrees

- cooperative sensors, wherein a target transmits information about its position;
- non-cooperative sensors. These kinds of sensors can indirectly sense a target, through either passively sensing an attribute of the target, or by actively deploying energy to seek out the target.

Surveillance methods that sense a cooperative target will usually employ a transponder method by which the target transmits information about its position. Such as the Traffic Collision Avoidance System (TCAS) relies on this method to discover another aircraft. Surveillance methods that sense non-transponder targets indirectly are considered non-cooperative sensing methods. A target is sensed and tracked, either (1) through passively acquiring information about the target (e.g., optical camera recording the reflected light, or acoustic sensor perceiving the target by passively listening); or (2) by actively deploying energy to seek out the target (e.g., radar which emits an electronic pulse and determine range and bearing by the angle of sensor and timing of the response, or laser range finder which emits infrared coherent light and detects reflections).

There are several strategies to manage the shortfalls of the sensor types. One could employ multiple sensors to cover a larger area, or multiple sensor types. One could reduce the detection and tracking requirement from fine resolution object tracking to area sensing (i.e., if there is anything detected in this sector, avoid the sector). One could simplify the avoidance reaction maneuvers, so that the sensing requirement is limited.

Frequency Selection

A very important design step is the selection of the frequency band in order to allow a reliable inter-vehicle communication in the various operational and topological scenarios on railroads.

The GSM-R is at about 900 MHz, which is a communication standard for data and voice based on GSM for European high-speed trains using base stations of sufficient height. In Japan, a band at 300 MHz is allocated for such applications, whereas in USA and Canada two bands at 160 MHz and 455 MHz are utilized. For Europe two suitable bands from 456-459 MHz and 460-470 MHz that were marked for railway communication services by the European Radio communications Committee (ERC).

The core feature of an ad-hoc network is to provide communication services without any infrastructure or centralized access point. There is no base station to coordinate packet transmissions. Since the direct train-to-train communication is intended to be used in specific networks, where the lines conditions should be taken into consideration, and of course the antennas are just mounted on top of the rail vehicles. That means, we face a much higher propagation loss in case of direct vehicle-to-vehicle communication.

3 System safety improvement by involving train-centric communication

In this chapter, we evaluate and compare the safety performance of the traditional ETCS-2 and the new system which added train-centric communication module on ETCS equipment. The new system is an enhanced movement authority system, and it is defined as MA+ in this thesis.

The necessity of involving train-centric communication system is discussed in section 3.1. Section 3.2 aims at evaluating the system safety of the MA+ system. In this section, the overall safety improvement ratio of the new system relative to ETCS-2 is calculated via MTBF with the assumption that the dangerous side failure rates of all equipment are the same. Moreover, we take the actual data from Chinese Wuhan-Guangzhou high-speed railway as an example, which shows there would be 24.04 times safety increase providing using of the new proposal. As involving the train-centric communication system, the train to train collision can be reduced. Hence, a collision failure model is presented in section 3.3.

3.1 Necessity of applying the train-centric communication

In 2010, European Directive mentioned that Intelligent Transportation System (ITS) would be one of the most important transportation technology that will improve transport safety. Further, developing a system that will establish the communication link between various parties (e.g. between a car and a train) is the main element of ITS. Nevertheless, the development of such technology is currently in progress. According to [35], statistics data has proven that traffic safety can be improved by developing the system that would help and

support vehicle drivers. This can be a system that assists vehicle drivers and train operators to avoid accidents [36].

Improving transportation safety requires numerous suitable methods. Reducing transitions to hazardous operations and increasing transitions to safer operations are the effective and practicable methods [37]. Limiting the movement violation is one of the efficient and innovative methods to reduce such accidents.

Different technologies have been applied to guarantee the safety of railway transportation. The latest ETCS-2, which is based on the fixed block principle, safely optimizes the maximum capacity of the rail network. The movement authority (MA) together with speed information and route data are transmitted to the vehicle via the Global System for Mobile Communications-Railway (GSM-R). However, the train detection and interval supervision remain in place at the trackside and sensors (axle transducers, accelerometer, and radar). The train can only passively move based on the MA, and no train to train communication is available.

Generally, the macroscopic physical movement embodies the following three aspects. As long as there is no modification in the fundamental character of the railway that involves trains moving along tracks, these three issues cannot be expected to change in the future [38].

- Scalar magnitude. The train movement is reflected as a position changing on the track. No moving in a period can be treated as a special movement with a magnitude 0.
- Vector direction. Modified by means of the switch, the train can move in different directions. It contributes to the vector direction varieties.
- Other objects, such as other trains and switches.

From a safety perspective, the movement under the scalar magnitude and vector direction should ensure the safety relationship between trains and switches. In current ETCS-2, these three factors are distributed in the interlocking system and the MA. As shown in Fig. 3.1, the interlocking details are invisible to trains.

To cover all these three aspects, the MA+ should receive the information of switches and trains. To obtain this information but without disturbing the original infrastructures, an additional communication channel, which is available for the direct connection of train-to-train and train-to-switch, is established.

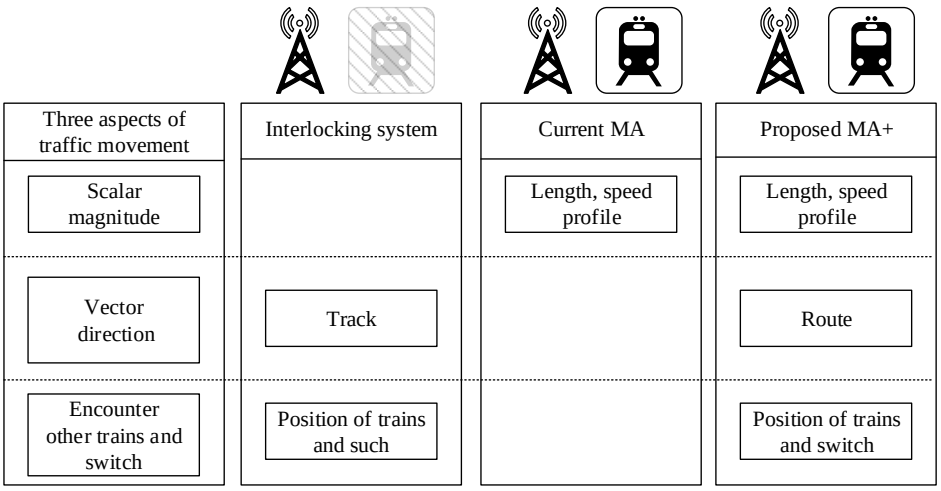


Figure 3.1: Movement information in different railway control policies

3.2 The overall system safety calculation of the MA+

In this section, the system safety evaluation carried out by analyzing the train safe distance interval and overall system safety. The numerical analysis is carried out by calculating the reliability of the average dangerous side failure probability. The result shows the system safety increases approximately linearly as the average cost of the wayside equipment relative to the trains.

Railway safety is an interdisciplinary research discipline [39]. Every new technology must at least provide an equivalent level of safety, comparable to the system it will improve or replace [14]. As shown in Fig. 3.2, the original risks of the system hazards are represented as a deep-blue line, and the green arrow line refers to the reduction of risks due to the application of the new system. Since each new system has a probability to fail, the risks originated from the new system are shown as a red arrow line in Fig. 3.2 [40]. Hence, the yellow line is the probability of hazards with the new system, which we should make it as low as possible, at least not higher than the original risks of the system.

According to EN 50126 / IEC 62278: Railway applications - The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS), the system safety can be indicated as the average probability/frequency of dangerous side failure of the safety system and/or function, which can be shown as:

$$\lambda_{us} = \mu * \lambda/h \tag{3.1}$$

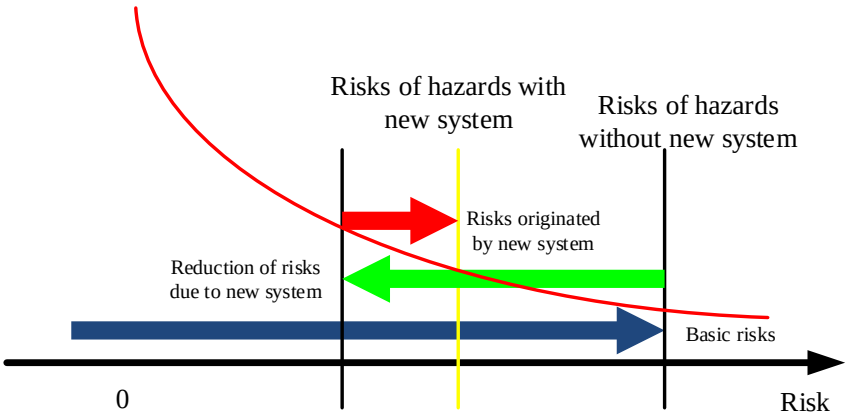


Figure 3.2: Risk of hazards with a new system involving

where λ is the failure rate of the system, it is often used to indicate the system reliability; λ_{us} is the failure rate of dangerous side, it is an expression of the system safety; μ is the coefficient of the dangerous side failure, it is often supposed to be a constant.

The importance of equation (3.1) is that it combines the system safety with reliability. Hence, the safety evaluation of the new proposal compared with ETCS-2 system can be alternated to the reliability comparison as shown in equation (3.2).

$$\frac{\lambda_{usW}}{\lambda_{usETCS}} = \frac{\mu_W * \lambda_W}{\mu_{ETCS} * \lambda_{ETCS}} = \varpi * \frac{\lambda_W}{\lambda_{ETCS}} \tag{3.2}$$

where λ_W, λ_{usW} and μ_W represent the failure, dangerous side failure, and the coefficient of the dangerous side failure rates of the whole system that with the MA+ module, respectively; λ_{ETCS}, λ_{usETCS}, and μ_{ETCS} represent the failure, dangerous side failure, and the coefficient of the dangerous side failure rates of ETCS-2 system, respectively; ϖ is the conversion coefficient of reliability to safety, and it is a constant.

Assuming that the coefficient of the dangerous side failure rates for the new system and the initial ETCS-2 system are nearly equal, which results in $\varpi \approx 1$. Hence, equation (3.2) can be rewritten as equation (3.3), which means the safety comparison result nearly equals to the reliability comparison result.

$$\frac{\lambda_{usW}}{\lambda_{usETCS}} \approx \frac{\lambda_W}{\lambda_{ETCS}} \tag{3.3}$$

Our proposal of the new system is a hot-standby redundant safe control structure. In this

structure, the ATP in ETCS-2 plays a major role while the MA+ module plays a backup or supervisory role. The system safety can be ensured as long as the ATP or the MA+ equipment operates reliably as designed, or does not fail to the dangerous side. The reliability block diagram of the whole system is shown in Fig. 3.3.

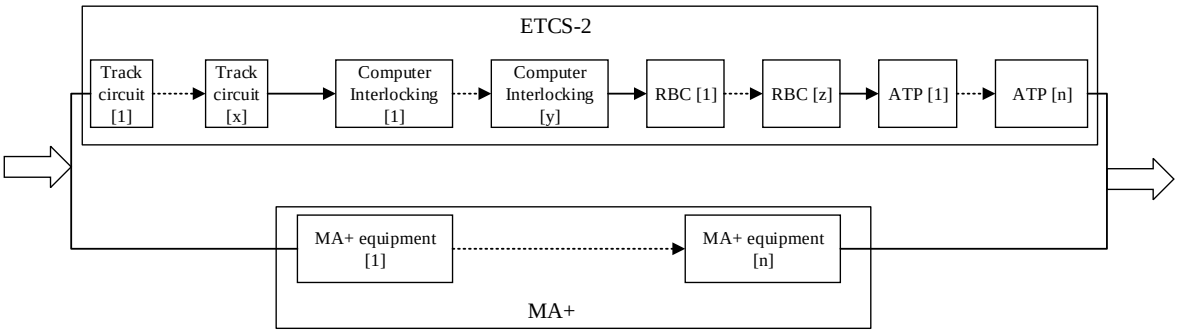


Figure 3.3: Reliability block diagram of the new system

According to the reliability theory, the reliability of the whole system $R_{W(t)}$ can be calculated as:

$$R_{W(t)} = R_{ETCS(\mathrm{t})} + R_{MA+(\mathrm{t})} - R_{ETCS(\mathrm{t})} * R_{MA+(\mathrm{t})} \tag{3.4}$$

where $R_{ETCS(\mathrm{t})}$ and $R_{MA+(\mathrm{t})}$ are the reliability of all the ETCS-2 and MA+ equipment, respectively.

$MTTF$ is often used to indicate the system reliability. The definition of $MTTF$ is the definite integral evaluation of the reliability function [41]. Given that the electronic equipment is often treated as an exponential distribution with a constant failure rate, $MTTF$ equals to the reciprocal of the failure rate λ_{t} as shown in equation (3.5).

$$MTTF = \int_0^\infty R_{(t)} dt = \frac{1}{\lambda_t}\, h \tag{3.5}$$

The reliability of typical ETCS-2 system depends on the track circuit (TC), the computer interlocking (CI), the radio block center (RBC), and the ATP module (ATP) work in

series [15]. Hence, the $MTTF$ of ETCS-2 is given by equation (3.6).

$$\begin{aligned} MTTF_{ETCS} &= \frac{1}{\lambda_{ETCS}} = \frac{1}{\lambda_{TC}+\lambda_{CI}+\lambda_{RBC}+\lambda_{ATP}} \\ &= \frac{1}{\sum_{i=1}^{x}\lambda_{TC[i]}+\sum_{j=1}^{y}\lambda_{CI[j]}+\sum_{k=1}^{z}\lambda_{RBC[k]}+\sum_{m=1}^{n}\lambda_{ATP[m]}}\ h \end{aligned} \tag{3.6}$$

where x, y, z, and n are the number of equipment for each type; λ_{TC}, λ_{CI}, λ_{RBC}, and λ_{ATP} are the total failure rates of each equipment type; i, j, k, and m represent the specific one of the equipment for each type, respectively.

Similarly, the reliability of new MA+ proposal depends on the preceding and the following train work in series, and every train on the line can be the preceding or following train of each other. Hence, the $MTTF$ of the MA+ equipment can be calculated as equation (3.7).

$$MTTF_{MA+} = \frac{1}{\lambda_{MA+}} = \frac{1}{\sum_{m=1}^{n}\lambda_{MA+[m]}}\ h \tag{3.7}$$

where $MTTF_{MA+}$ is the total MTTF of the all the equipment; λ_{MA+} is the total failure rate of the MA+ proposal. As there is a MA+ equipment for one ATP, its number is n as well.

Combining the equations (3.4) - (3.7), we can obtain the $MTTF$ of the whole system as:

$$\begin{aligned} MTTF_W &= \int_0^\infty R_{W(t)}dt \\ &= \int_0^\infty [R_{1(\mathrm{t})} + R_{2(\mathrm{t})} - R_{1(\mathrm{t})} * R_{2(\mathrm{t})}]dt \\ &= \frac{1}{\lambda_{ETCS}} + \frac{1}{\lambda_{MA+}} - \frac{1}{\lambda_{ETCS}+\lambda_{MA+}}\ h \end{aligned} \tag{3.8}$$

In the case that the equipment in signaling system failure leads to dangerous side with a ratio of 1000:1, and ATP equipment reaches SIL4 with a failure rate $10^{-9}/h$. Hence, $MTTF$ of the track circuit, computer interlocking, radio block center, and ATP are required to be 10^5 h, so all the failure rates of these equipment will be the same as $10^{-5}/h$ [15]:

$$\lambda = \lambda_{TC[i]} = \lambda_{CI[j]} = \lambda_{RBC[k]} = \lambda_{ATP[m]} = \lambda_{MA+[m]} = 10^{-5}/h \tag{3.9}$$

Additionally, the MA+ equipment is very similar with the onboard ATP, it is assumed that

its failure rate equals to ATP. Hence, the reliability and safety increase ratio r is calculated as shown in equation (3.10).

$$
\begin{aligned}
r &= \frac{MTTF_W}{MTBF_{ETCS}} = \frac{\frac{1}{\lambda_{ETCS}} + \frac{1}{\lambda_{MA+}} - \frac{1}{\lambda_{ETCS}+\lambda_{MA+}}}{\frac{1}{\lambda_{ETCS}}} \\
&= \frac{\frac{1}{(x+y+z+n)*\lambda} + \frac{1}{n*\lambda} - \frac{1}{(x+y+z+n)*\lambda + n*\lambda}}{\frac{1}{(x+y+z+n)*\lambda}} \\
&= \tau + 2 - \frac{\tau+1}{\tau+2}
\end{aligned}
\tag{3.10}
$$

where τ represents the average cost of the wayside equipment, which can be calculated as equation (3.11); x, y, z, and n are the number of equipment for track circuit, computer interlocking, radio block center, and ATP, respectively. Obviously, a line with a large number of wayside equipment and few trains costs much more than a line with few wayside equipment but lots of trains.

$$
\tau = \frac{x+y+z}{n} \tag{3.11}
$$

The limitation of τ can be $(0, +\infty)$. Two extreme values indicate two scenarios: a line with no wayside equipment, and a line with no trains, respectively. So the reliability and safety increase ratio r can be from 1.5 to positive infinity according to equation (3.10). The fewer trains and the more wayside equipment in the line, the greater safety performance improvement through applying the MA+ proposal.

Taking the Chinese “7.23 Railway Accident" as a case study to qualitatively evaluate the proposal. It is an accident resulted from the failure of the wayside signal system. A rear-end collision occurred between two trains on the Yongwen High-Speed Line in China on July 23, 2011. After the track circuit was failed after struck by lightning, the wayside Train Control Center (TCC) subsystem continuously outputting erroneous permit signal. As the track circuit error, the preceding train executed an emergency stop, then run at speed lower than $20km/h$ in on-sight mode. Hence, the preceding train did not leave the track section timely. Meanwhile, the following train was permitted to enter the section as the TCC fails to locate the preceding train. Although the driver was informed the existence of the preceding train by the Operations Control Center (OCC) operator 30 seconds before the accident, it was too late to stop the following train. The accident resulted in 40 deaths and more than 170 serious injuries. From the cause analysis of the accident and principle of the new MA+

proposal, we can see that if it were equipped with the MA+ module, the accident would be avoided by the MA+ module.

To evaluate the proposal objectively and quantitatively, we took Chinese Wuhan-Guangzhou high-speed railway as another case, where the values of x, y, z and n are 2600, 13, 9, and 114 respectively. We can easily calculate and get the result that $r = 24.04$. It means the reliability and safety would increase 24.04 times by using MA+ proposal in this line, which can bring a significant improvement in the quality of service.

3.3 Train to train collision failure model

Train collision is one of the most serious accidents in railway transportation. There are three different potential train to train collisions: head-on, rear-end, and flank, as shown in the Fault Tree Fig. 3.4.

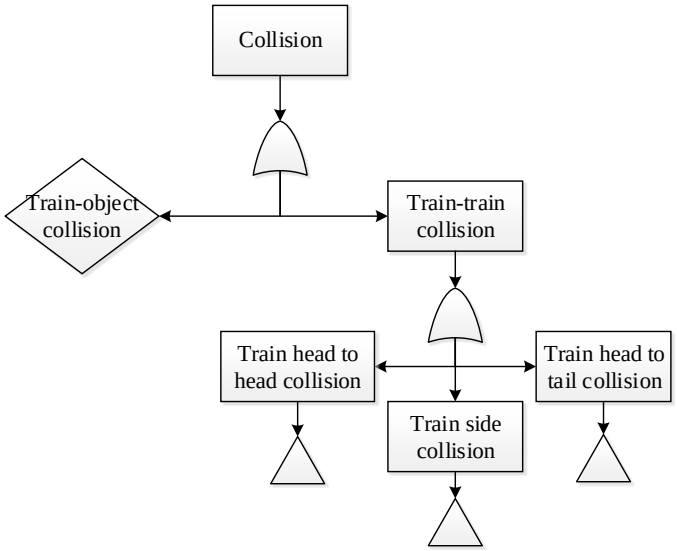


Figure 3.4: Fault Tree of the collision failure model (adapted from Hartong, 2011)

Normally, with the implementation of double track railway, head to head collision is relatively low. Additionally, the track geometry is carefully analyzed before the railway laying. Hence, the train head to tail collision is the constant concern in railway transportation. It is the consequence of three possible events as shown in Fig. 3.5: failure to establish correct routes, failure to enforce routes, and trains over speed for the track conditions resulting in overrunning limits of authority [14].

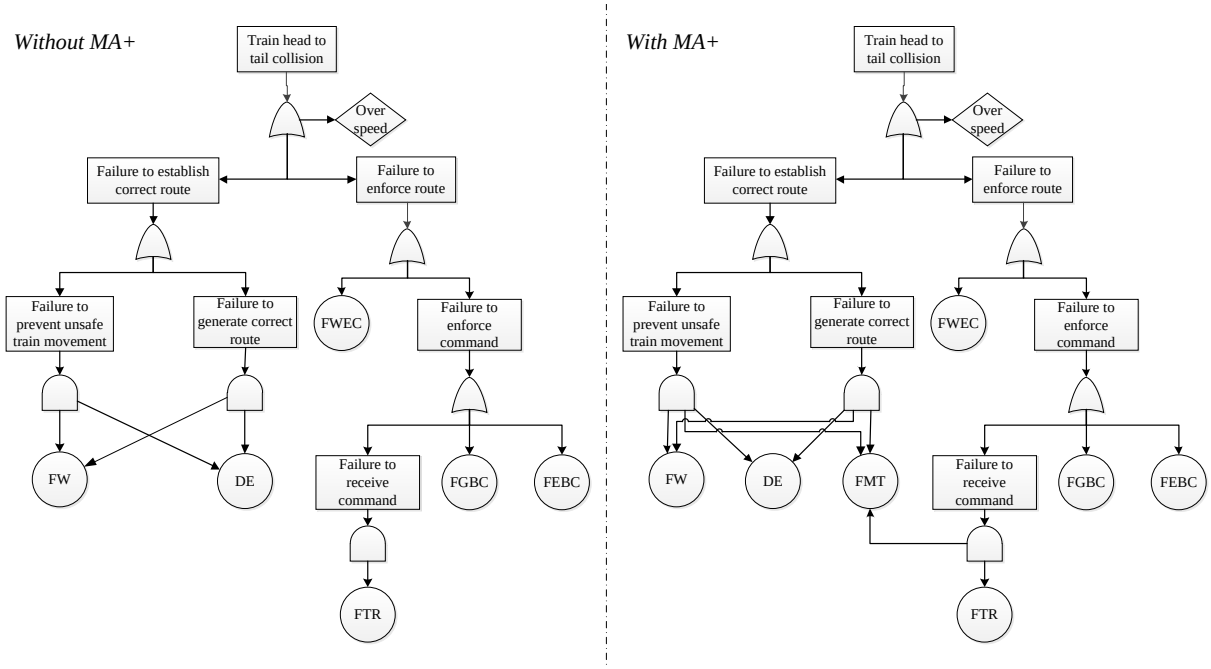


Figure 3.5: Fault Tree of the train head to tail collision model

Failures to establish correct route are attributed to failures of expressing correct movement. Failure of wayside equipment (FW) and unidentified by the dispatcher (DE), which allows an unsafe movement or generates false signals, leads to the collision, as the Wenzhou line accident a case study. This accident resulted from the failure of the wayside signal system. Although the driver was informed the existence of the preceding train by the Operations Control Center (OCC) operator 30 seconds before the accident, it was too late to stop the following train.

The collision is arising from the failure of human factors, which is a leading cause of the accidents. 86% of train crashes and 33% of derailments are related to human factors [42]. When the wayside equipment and interlocking are out of service, the train dispatcher takes responsibility for ensuring no two trains are authored to occupy the same track at the same time. The train crews are responsible for obeying the commands. Hence, failures of the dispatcher (DE) sets correct routes or allows unsafe train movement to happen, and then train collisions will occur. For instance, the dispatcher was to blame for trains head-on crash in Bavaria, Germany.

Failures to enforce a correct route are attributed to the failures of command transmission and implementation. Failure of the wayside equipment to execute the command (FWEC), which allows an insecure movement, can be the reason for the system failed to display the correct route to the crew. Signal information is transferred from the wayside equipment to

the train, and then to the onboard equipment outputs command. Failures of preventing against conflicting results could be a failure of receiving the command transmission (FTR). The control center generates correct routes and signals, but the train failed to enforce it. Hence, collisions can be the result of a failure of the transmission system to convey the correct route to the train, failure of the onboard system to generate appropriate braking commands (FGBC) to protect against conflicting routes, or failure of the braking system (FEBC) to correctly implement the braking commands.

When applying MA+ into existing train systems, the collision failure model will be modified. In the failure model, FMT is used to represent the failure of MA+. Since MA+ can independently obtain the switches' and forward trains' positions, even though the wayside equipment is out of service or the dispatcher sets an incorrect route. Hence, it will prevent unsafe movement caused by wayside equipment and dispatchers. Additionally, as an extra communication link, the MA+ can avoid accidents caused by failures of the command transmission.

4 Enhanced movement authority system (MA+)

MA+ combines advantages of the train-centric communication with current movement authority mechanisms, as shown in Fig. 4.1. The aim of this chapter intends to provide researchers the train-to-train distance measurement system (TTDMS), which is the main component of MA+, and carries out the train distance measurement and data communication.

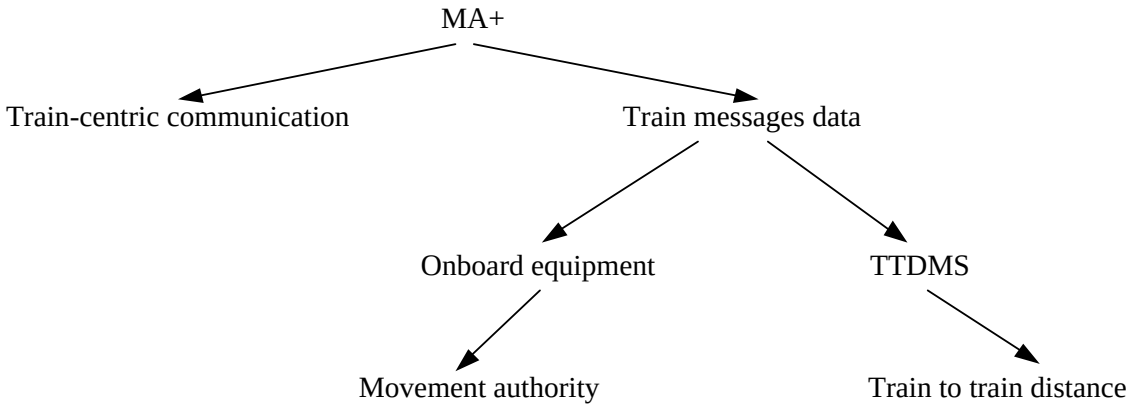


Figure 4.1: components of MA+

Varying from the systems mentioned in publications [25] and [26], the MA+ requires no additional position and speed measurement system. The essential information is collected by using the internal wireless Packets generated from ETCS's Automatic Train Protection (ATP) system. The external data exchange is implemented via a train-centric communication unit. Importantly, in order to maximize the utilization of this train-centric communication unit, we applied distance estimation based on the Time of Arrive (TOA). This strategy can serve as a backup distance estimation method for the MA+.

The remainder of the chapter is organized as follows. Section 4.1 introduces the MA+ basic structure and algorithm. Section 4.2 presents the principle of TTDMS. Section 4.3 proposed a prototype TTDMS, and its reliability is estimated. Section 4.4 discussed two strategies of train safe distance data acquisition. Section 4.5 calculates the safe distances of traditional ETCS and the MA+ proposal and compares mathematically to show that the new proposal in this chapter can work correctly and safely without disturbing the normal control process of the ETCS system.

4.1 System structure and algorithm of MA+

Based on the technology trends, the train control system should rid of the proportion of ground faculties, and give trains more individual initiative than the past. Some projects based on the train-centric communication are carried out in these years [24–26]. In principle, the MA+ is very similar to the Traffic Alert and Collision Avoidance System (TCAS) in aeronautical transport.

In order to obtain the extra information without disturbing the original infrastructures, an additional communication channel is established. It is available for direct connection between train-to-train and train-to-switch. In the designed MA+, each train broadcasts its position and movement authority as well as other required information to all other trains in the area. The MA+ has to receive the information of approaching switches and encountering trains. In the MA+, new train-centric communication architecture will be installed in each train, and the switch announcement architecture broadcasts the switch location and situation repeatedly. Therefore, the MA+ combines advantages of the train-centric communication and current movement authority mechanisms. Such an infrastructure-less collision avoidance overlap system has many advantages. For instance, the MA+ can serve as the supplement of GSM-R to enhance safety. Additionally, during the MA+ communication process, a backup distance measurement that based on Time of Different Arrive method is possible.

The structure of MA+ is indicated in Fig. 4.2. In the MA+, a new *train-centric communication unit* is required to install in each train, and the *switch announcement unit* broadcasts the switch's location and status repeatedly. In the *train-centric communication unit*, the *transceiver unit* is in charge of exchanging train location between different *train-centric communication unit*s, and receiving the information from *switch announcement unit*s. The *MA+ algorithm* analyzes the received information both from other *train-centric communication unit*s and the ATP equipment, and outputs results. The *digital map* provides the description of all the railway-related data including the infrastructure (the railway tracks,

stations, signals, line topology, and length, etc.), the parameters of the rolling stocks and other control equipment, the timetable (schedule), and so on [43]. The *TTDMS* can carry out the distance interval measurement as a backup strategy, more details are discussed in the following subsection.

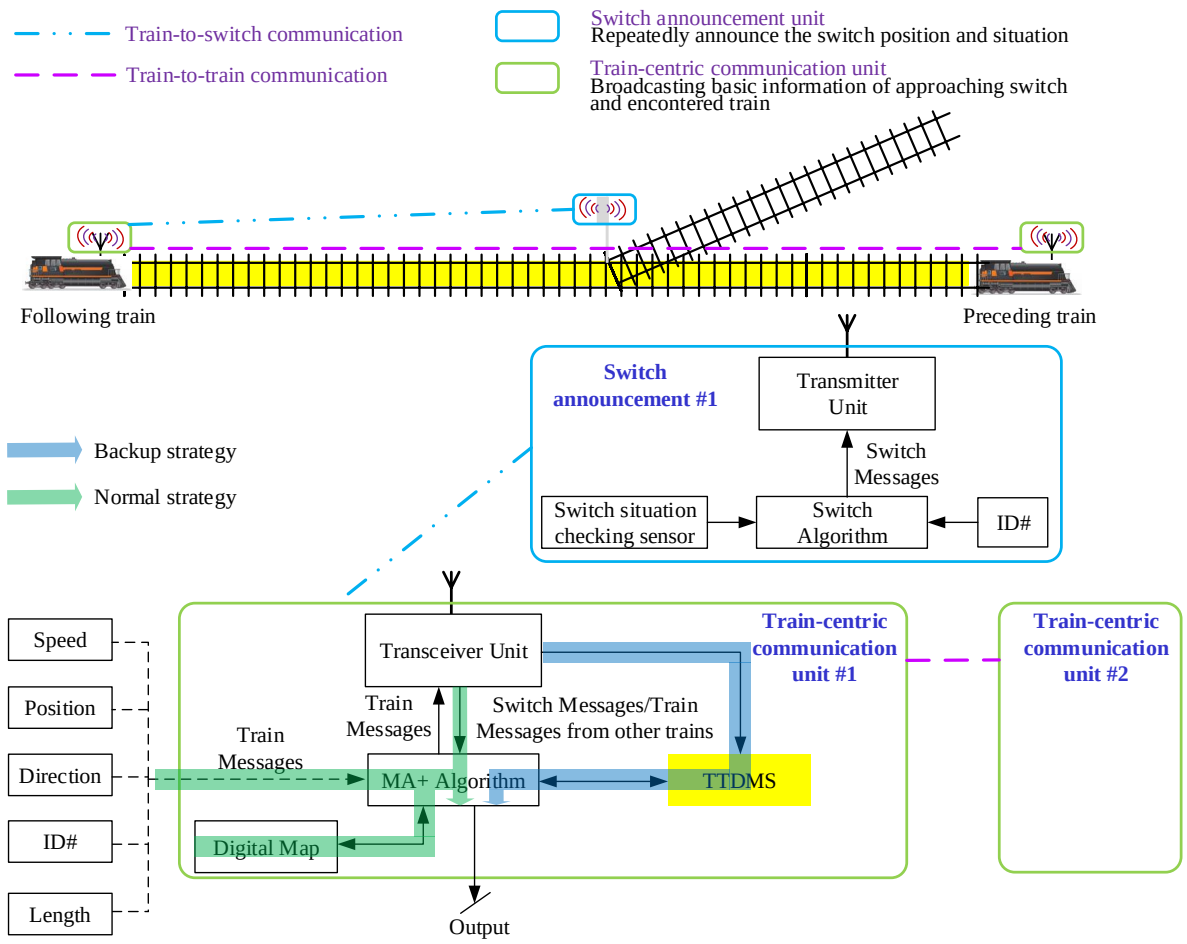


Figure 4.2: Structure of MA+

As presented in section 3.1, the MA only contains the train interval information but lacks the information of the interlocking and preceding train's position. Hence, the MA+ proposed in this chapter intends to monitor the train distance interval based on the moving block principle, and to provide interlocking information based on a digital map. The performance of MA+ highly depends on the resolution of the localization used. Hence, two strategies are proposed for the acquisition of train distance interval data:

- Normal strategy: in order to distinguish each train on the tracks based on position, the Train Messages (`TM`) and digital map, which can be abstracted from the onboard

equipment, are involved in the high correctness distance estimation.

- Backup strategy: when the onboard equipment or its connection with RBC is abnormal, the train-centric communication unit can estimate the train distance interval based on the TTDMS, which uses the TOA method to calculate the distance.

Taking the operation scenario in Fig. 4.2 as a case study. This scenario splits into two communication parts: train-to-switch and train-to-train. The train-to-switch communication link permits the train obtaining the Switch messages, which involves the switch's position and status. The train-to-train communication exchanges the `TM` between the following and preceding trains. The `TM` including train's speed, position, direction, ID, and length can be obtained from the onboard equipment, and be transferred to other trains.

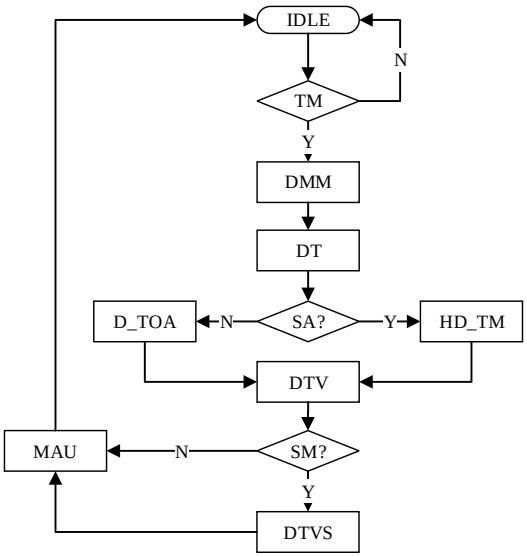


Figure 4.3: Flow chart of the MA+ algorithm

The logical model of the *MA+ algorithm* can be described as shown in Fig. 4.3. The MA+ starts from the `IDLE` mode. After obtaining and decoding the basic information from the `TM`, the MA+ matches the digital map (`DMM`) and obtains the track data (`DT`). The following train detects the preceding train and calculates the train distance interval based on the `TM` (`HD_TM`) when the onboard equipment operates functionally (`SA`). If the `TM` from the preceding train is not available, the TOA distance estimation unit can implement the distance measurement (`D_TOA`). Afterwards, the MA+ obtains the information of the track and vehicles (`DTV`). In the following steps, it detects the front switches (`SM`). If there is not switch in the detection range, the MA+ update the MA+ (`MAU`). Otherwise, the MA+ turns into the corresponding

mode, which involves digital map, track data, vehicles, and switch information (DTVS), then updates the MA+.

4.2 System principle of TTDMS

In this section, the principle and operation model of the TTDMS are described. The distance between two trains is calculated based on the TOA. The system application scenarios and block diagram are presented.

Physical space separation is a fundamental safety protection for trains. The primary goal is to obtain distance information between two trains in time. The mechanism of obtaining distance information is to determine a train's absolute position, for instance with Global Navigation Satellite System (GNSS), balise, or track circuit. Radar, speed sensors, and spread spectrum communication can place the train's relative position.

The TTDMS is based on the Time of Arrival (TOA) method to get distance information [8]. Importantly, it does not rely on other systems such as Global Position System (GPS), GLONASS, and so on. The information can be directly transferred between trains without involving the Centralized Traffic Control (CTC). Trains equipped with the TTDMS have a "See and Avoid" ability, which is already widely applied in aviation areas [44].

The technology applied in this TTDMS is using the time duration of wireless signals transfer in the air between the sender and the receiver to calculate the distance [45]. In this chapter, the TOA calculation relies on the characteristics of spread spectrum communication. The details of TOA estimation can be found in section 4.4. The time consumed by signal transmission can be captured using autocorrelation of pseudorandom noise (PRN) code [46]. With PRN code involved, the signal sequence has an autocorrelation attribution. Here, based on the TOA and PRN theories, radio transmission characteristics can be used to obtain the distance between two trains. At the same time, necessary information can be added to the communication code, such as train ID, velocity, and other control information.

Similar technologies have been applied in different fields. Some other organizations also have research in this technology. For example, in German Aerospace Center a Railway Collision Avoidance System was proposed by Prof. Dr. Thomas Strang and his team [26, 47]. Other examples are the Advanced Automatic Train Control (AATC) system used in San Francisco [48], the Enhanced Position Location Reporting System applied in the U.S. Army [49], and a train-centric CBTC solution putted forward by Alstom [25].

The TTDMS application scenarios and general process are shown in Fig. 4.4. It starts from surveillance mode, and then the following train tries to detect nearby trains equipped with TTMDS. After the communication is established, the following train sends a sequence with a PRN code sequence to the detected train, and the detected train feeds back the sequence after it has received it. The system analyzes the TOA information and generates results which involve, but are not limited to, distance and alarm information.

Since the TTDMS obtains the distance information based on TOA, the train-centric communication condition is important. The communication process involves connection establishment, transmission error, and disconnection. All these factors could cause an impact on calculation results. The system communication part provides distance measurement information for the TTDMS, the approaching information of other nearby trains is available to the following trains, as shown in Fig. 4.4.

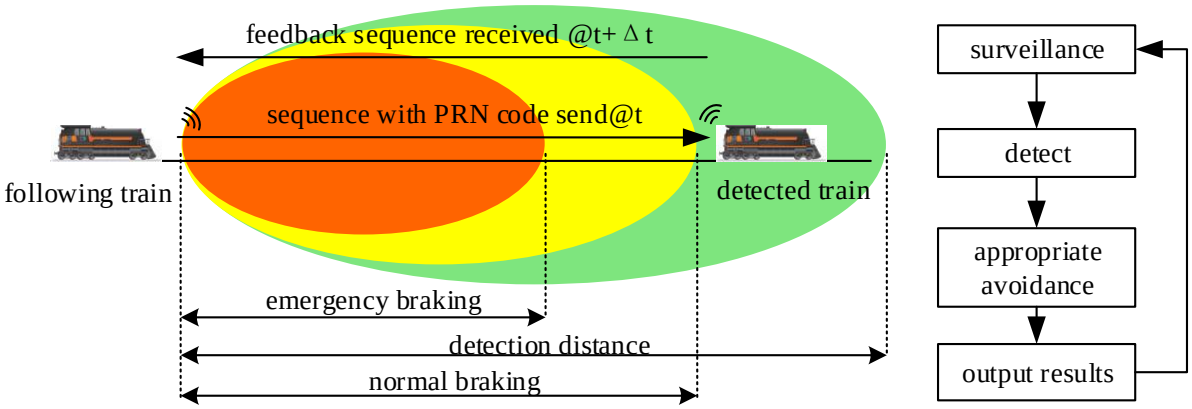


Figure 4.4: Application scenarios and general process

By applying the TTDMS, each train can have an overview of its surrounding environment, even though the ATP has lost the connection with the train control center. The train can still detect the distance between nearby trains, at the same time other trains can also detect it. The TTDMS can provide essential information for its service train, no matter whether other onboard systems are under normal operation or failed state. The TTDMS can work as an assistance system by providing essential and useful data for the service train. As a result, the TTDMS can improve the railway operation safety in some scenarios.

The TTDMS's application scenarios and block diagram are depicted in Fig. 4.5. The system starts from surveillance mode, and then the following train tries to detect nearby trains equipped with TTDMS. After the communication is established, the following train

sends a sequence with PRN code to the detected train, and the detected train re-sends the sequence after its reception. The system analyzes the TOA information and outputs results, which involve amongst others the distance and alarm information.

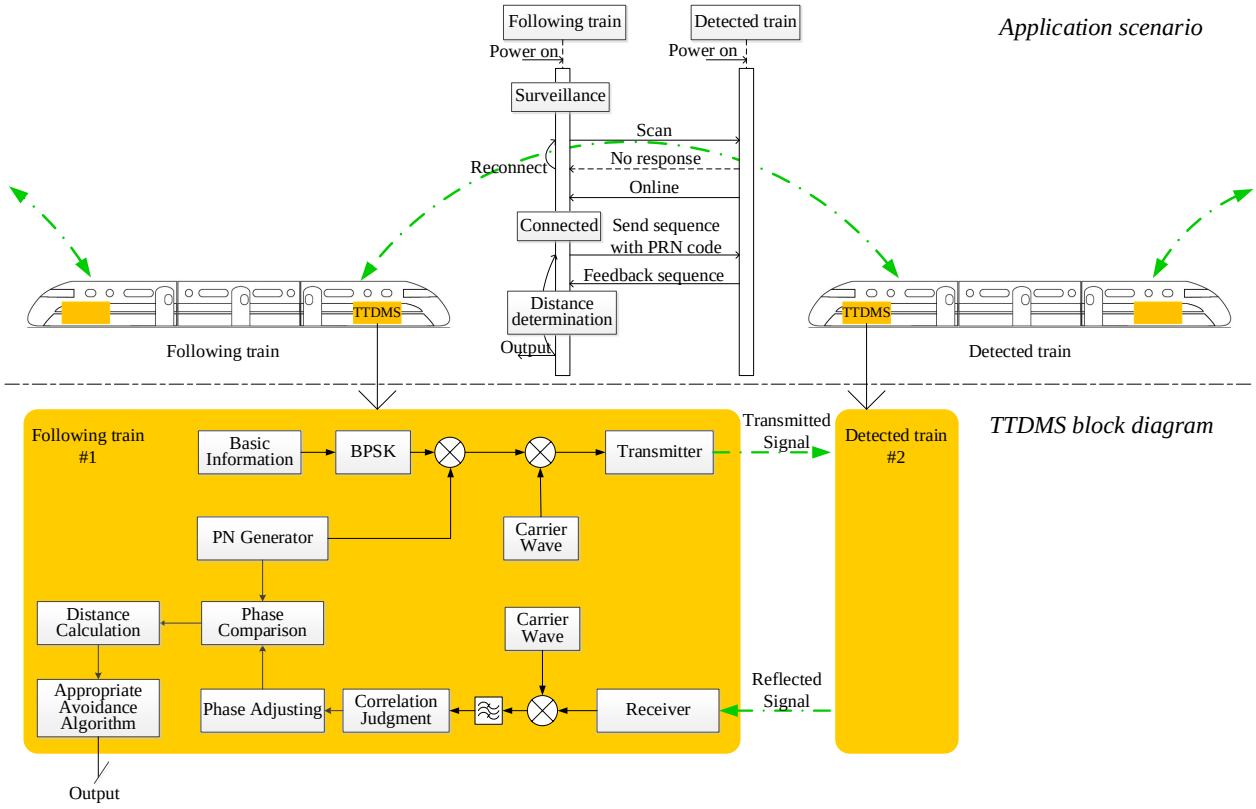


Figure 4.5: Application scenarios and block diagram of TTDMS

4.3 Actual TTDMS reliability estimation

To further the system principle into the engineering implementation, we proposed a prototype machine, which is applied to do actual measurements in metro lines. The prototype machine is as shown in Fig. 4.6. It operates at frequency 1.2 GHz with a peak power <10 mW. Because of the antenna power limitation and hardware ability, the experimental data was obtained in straight road scenarios at a distance of 1000 meters and 500 meters in tunnel scenarios. Fig. 4.7 is the system's application scenario in a tunnel.

Hundred percent reliable railway systems do not exist. System failures cannot be avoided. The overall failure rate of a system can be represented by λ_{system}, it should be as low as possible. The consequences of the system's failure can be divided into two groups, hazardous

Figure 4.6: TTDMS prototype machine

Figure 4.7: TTDMS prototype machine application scenario in tunnel

and safe faulty states. λ_h and λ_s are used to represent the failure rates of these two faulty states, respectively.

Failure rates of electronic components can be divided into three periods, as shown in Fig. 4.8. It includes early failures, random failures, and wear out failures. Each section is described with its Weibull distribution as $\lambda(t) = \frac{b}{T} \cdot \left(\frac{t}{T}\right)^{b-1}$ [41]. Here, b is the shape parameter of

failure slope. T is the characteristic lifetime. t is the service time. For electronic components, a burn-in process is required, which makes the components enter into the random failures period. At this period, the shape parameter is 1.0. Hence, the failure rates are constants. The manually burning process is shown in Fig. 4.9.

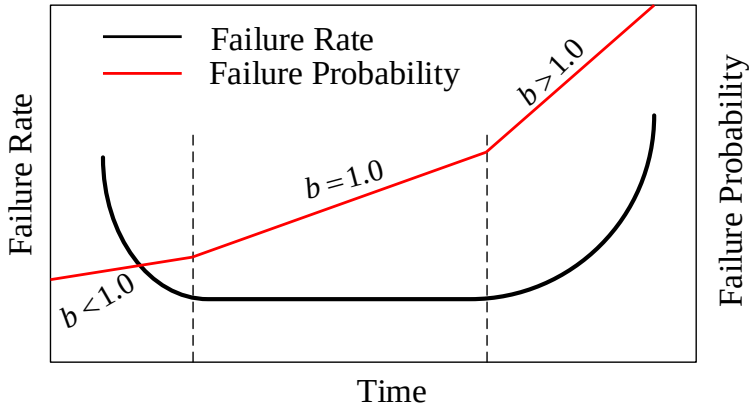


Figure 4.8: Failure rates and Weibull parameter b

Figure 4.9: Manually burning

For the TTDMS, the CPN model will reflect the main function of the TTDMS. The distance measurement unit implements the distance measurement. As shown in Fig. 4.10, the whole TTDMS involves different units: Power supply Module (DC), Baseband Module (BM), Distance Measurement Module (DM), Input and Output Module (IO), and Display Module (DIS).

At this period the failure rates are constants [41]. The failure rate (λ_{unit}) of an unit mentioned

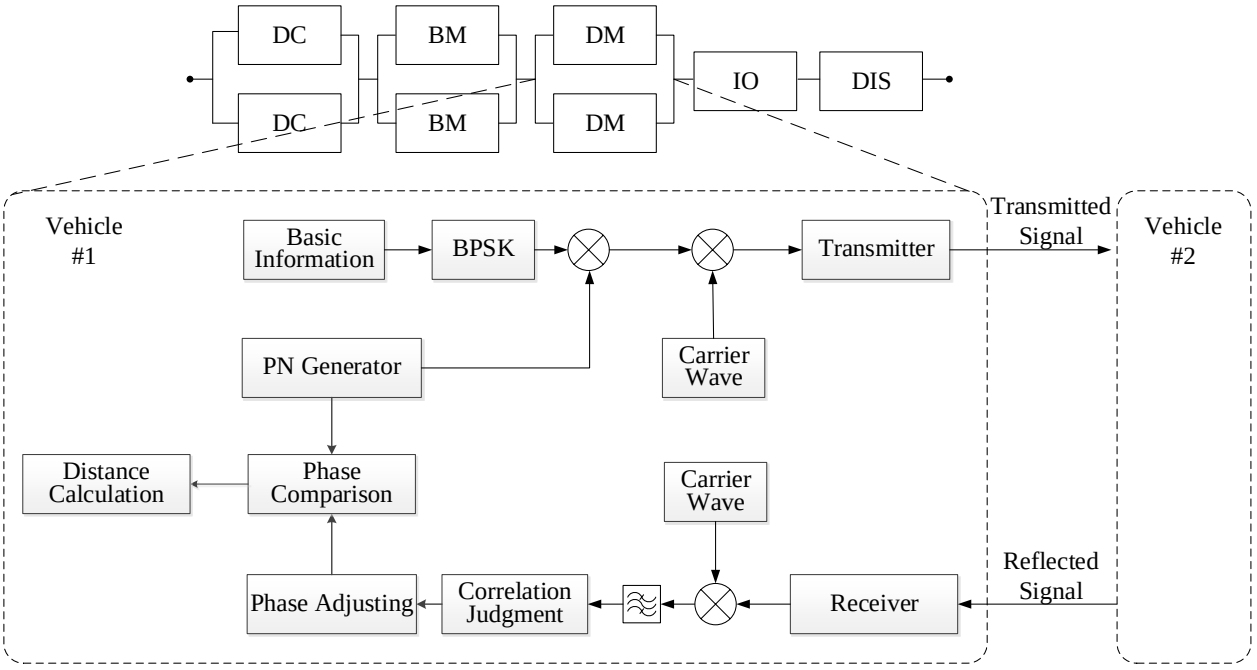


Figure 4.10: Structure of distance measurement unit

above is shown in equation (4.1).

$$\lambda_{unit} = \sum_{i=1}^{n} N_i(\lambda_{Gi}\pi_{Qi}) \quad h^{-1} \tag{4.1}$$

where λ_{Gi}, π_{Qi}, and N_i are the general failure rate, quality, and quantity of the electronic component i, respectively. The calculations of the reliability prediction of electronic equipment can be carried out according to MIL-HDBK-217. Based on the MIL-HDBK-217, the failure rates of different modules are calculated.

Table 4.1: Failure rates values

Failure rate type	Numerical value $\times 10^{-6} h^{-1}$	MTBF (h)
λ_{DC}	18.7	53225
λ_{BM}	20.5	48852
λ_{DM}	19.3	51803
λ_{IO}	17.2	58102
λ_{DIS}	22.3	44782

For a series system with constant device failure rates, Mean Time To Failure (MTTF) is

defined as

$$MTTF_{SERIES} = \int_0^\infty R_{SERIES}(t)\,dt = \int_0^\infty [R_1(t) \times R_2(t) \times \cdots \times R_i(t)]dt$$
$$= \int_0^\infty e^{-\left(\sum_{i=1}^{n} \lambda_i\right)} dt = \frac{1}{\sum_{i=1}^{n} \lambda_i} \tag{4.2}$$

where R_i and λ_i are the reliability and failure rate of device i, respectively.

Consider a two device parallel system that has devices A and B , the $MTTF_{PARALLEL}$ is given by:

$$MTTF_{PARALLEL} = \int_0^\infty R_{PARALLEL}(t)\,dt = \int_0^\infty [R_A(t) + R_B(t) - R_A(t) \times R_B(t)]dt$$
$$= \int_0^\infty \left[e^{-\lambda_A t} + e^{-\lambda_B t} - e^{-(\lambda_A+\lambda_B)t}\right] dt = \frac{1}{\lambda_A} + \frac{1}{\lambda_B} - \frac{1}{\lambda_A + \lambda_B} \tag{4.3}$$

Hence, the reliabilities of DC, BM, DM, IO, and DIS modules are shown in equations (4.4), (4.5), (4.6), (4.7), and (4.8), respectively.

$$R_{DCModule}(t) = 2e^{-\lambda_{DC} t} - e^{-2\lambda_{DC} t} \tag{4.4}$$

$$R_{BMModule}(t) = 2e^{-\lambda_{BM} t} - e^{-2\lambda_{BM} t} \tag{4.5}$$

$$R_{DMModule}(t) = 2e^{-\lambda_{DM} t} - e^{-2\lambda_{DM} t} \tag{4.6}$$

$$R_{IOModule}(t) = e^{-\lambda_{IO} t} \tag{4.7}$$

$$R_{DISModule}(t) = e^{-2\lambda_{DIS} t} \tag{4.8}$$

Substituting the reliabilities into the $MTTF_{SERIES}$ definition yields, the MTTF of the overall system is given by:

$$\begin{aligned} MTTF_{system} &= \int_0^\infty [R_{DC}(t) \times R_{BM}(t) \times R_{DM}(t) \times R_{IO}(t) \times R_{DIS}(t)]dt \\ &= 16908 \quad h \end{aligned} \tag{4.9}$$

The core mission of the TTMDS is implementing the distance measurement. It is carried out by the DC, BM and DM units. Hence, the failure rate of the unsafe state is:

$$\lambda_h = \frac{1}{\int_0^\infty R_{DC}(t) \times R_{BM}(t) \times R_{DM} dt} = 2.79 \times 10^{-5} \quad h^{-1} \tag{4.10}$$

To specify this equipment's adequate reliability level, we refer to the Safety integrity level (SIL), as shown in Table 4.2. This TTDMS equipment reaches SIL 4 in the low-demand mode of operation, which was also as a result of parallel structures.

Table 4.2: Quantitative SIL requirements

Safety Integrity Level	Low-demand mode of operation	High-demand mode of operation
	Average Probability of Failure on Demand (PFD_{avg})/activation	Probability of dangerous Failure per Hour (PFH)/h
SIL 4	$10^{-5} \leq PFD_{avg} < 10^{-4}$	$10^{-9} \leq PFH < 10^{-8}$
SIL 3	$10^{-4} \leq PFD_{avg} < 10^{-3}$	$10^{-8} \leq PFH < 10^{-7}$
SIL 2	$10^{-3} \leq PFD_{avg} < 10^{-2}$	$10^{-7} \leq PFH < 10^{-6}$
SIL 1	$10^{-2} \leq PFD_{avg} < 10^{-1}$	$10^{-6} \leq PFH < 10^{-5}$

It implies that the TTDMS can provide a considerable hardware reliability in actual implementation.

4.4 Train safe distance interval data acquisition of MA+

This chapter calculates the safe distance interval between two following trains to show its feasibility. Calculation of the train safe distance interval by MA+ is somewhat like the way of moving block system. If the interval between two following trains is shorter than the safe range due to any reasons, such as an equipment failure or unsafe interaction between ATP and the train, or the RBC dysfunctionality, the MA+ equipment will output a control instruction to the driver or execute brake action directly. As a result, the MA+ can enhance

the safety of train tracking without influencing the train operation under the protection of ATP equipment.

In the MA+, there are two strategies to carry out the calculation of the train safe distance interval. As illustrated in Fig. 4.3, the first one is based on `TM` and the *digital map* (`HD_TM`); the second one is based on the *TOA distance estimation* (`D_TOA`).

4.4.1 Normal strategy: high correctness distance estimation

When the ATP onboard equipment operates functionally, it can generate the train's position with high precision. In ETCS-2, the train localization is based on the absolute position information from balises and the odometry onboard. The uncertainty of balise's position and geographic data is given with typically 5 m; odometry error resulting from tachometer is provided with typically 1% of the distance from the last balise [21].

MA+ algorithm receives and decodes all the data coming from ATP onboard equipment and *transceiver unit*. Fig. 4.11 indicates the `TM` exchanging process. The `TM` is obtained from the Packets of radio messages. Table 4.3 indicates partial definitions of Packets, which are essential in composing the MA+. Packet 0 contains details of train position information. The train's position, speed and driving directions are available in variables *D_LRBG*, *V_TRAIN* and *Q_DIRTRAIN*, respectively. Variable *L_TRAIN* in Packet 11 indicates the real length of the train. *NID_OPERATIONAL* is the train's running number in Packet 5.

Table 4.3: Packet 0, 11 and 5 (partial)

Variable	Comment	Direction of the information
Packet 0		
D_LRBG	Distance between the LRBG and the train	Train to track
V_TRAIN	Actual train speed	
Q_DIRTRAIN	Direction of train movement	
Packet 11		
L_TRAIN	the absolute real length of the train	Train to track
Packet 5		
NID_OPERATIONAL	Train running number ID	Train to track

In the MA+ system, the following train can obtain the location detail of the preceding train through the *transceiver unit*, and calculate the train distance interval after matching with the *digital map*.

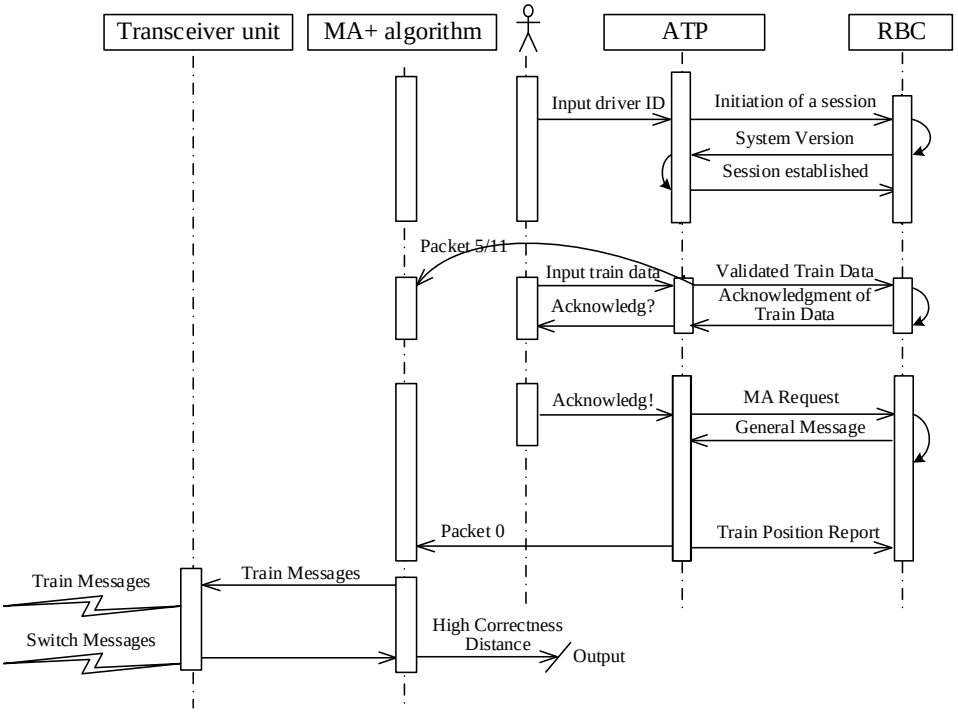


Figure 4.11: `TM` exchanging process

4.4.2 Backup strategy: TTDMS

When the ATP onboard equipment or its connection with RBC is abnormal, the TTDMS can estimate the train distance interval based on the TOA method [50]. The TOA method calculates the train distance interval by measuring the time duration (τ) of wireless signal transfers between two trains. Obtaining the time consumed by signal transmission is the first step to calculate the distance, since

$$L = \frac{\tau}{2} \times C \quad m \tag{4.11}$$

where L represents the distance from a following train to a preceding train. C is the velocity of light, $C = 3 \cdot 10^8\ m/s$.

The autocorrelation of the PRN code is used to capture the τ, which is hard for the hardware to measurement directly. The PRN code generation process is not introduced in this paper, reference [51] introduces different spreading codes.

For truly random sequences $u(t)$ and $w(t)$, which are infinite and statistically independent,

the correlation function is defined as:

$$R_{uw}(\tau) = \int_{-\infty}^{\infty} u(t)\, w^*(t-\tau) dt \tag{4.12}$$

It is not realizable to generate a truly random chip sequence in engineering, but the periodic PRN code provides a close approximation. For instance, the maximal-length $m-$sequences with N_c chips, which can be generated through an $n-$element shift register [52]. The PRN code generation process is not introduced in this chapter, however, [51] introduced different spreading codes. The correlation function of the $m-$sequence is given by:

$$R(\tau) = \begin{cases} 1 - \frac{N_c+1}{N_c}\frac{|\tau|}{T_c}, \tau = 0 \quad (mod N_c) \\ -\frac{1}{N_c}, \tau \neq 0 \quad (mod N_c) \end{cases} \tag{4.13}$$

$s(t)$ and $x(t)$ are the signals transferred from the following train and received by the detected train, respectively.

$$x(t) = s(t) + m(t) \tag{4.14}$$

where $m(t)$ and $n(t)$ are the channel noise, which are statistically independent.

$y(t)$ is the signal transferred from the detected train and received by the following TTDMS, thus

$$y(t) = s(t-\tau_0) + n(t) + m(t) \tag{4.15}$$

where τ_0 is the time difference between the original and received signals. The correlation function of $s(t)$ and $y(t)$ is given by

$$\begin{aligned} &R_{sy}(\tau) = E[y(t+\tau) \cdot s(t)] = \\ &E[s(t+\tau-\tau_0) \cdot s(t)] + E[s(t) \cdot n(t+\tau)] + \\ &E[m(t+\tau) \cdot s(t)] \end{aligned} \tag{4.16}$$

As channel noise $n(t)$, $m(t)$ and the original signal $s(t)$ are statistically independent. Hence,

$$E[s(t) \cdot n(t+\tau)] = 0 \tag{4.17}$$

$$E\left[m\left(t+\tau\right)\cdot s\left(t\right)\right]=0 \tag{4.18}$$

the equation (4.16) can be simplified as:

$$R_{sy}\left(\tau\right)=E\left[s\left(t+\tau-\tau_0\right)\cdot s\left(t\right)\right]=R_s\left(\tau-\tau_0\right) \tag{4.19}$$

The time offset of the signal is shown in Fig. 4.12. The original signal transferred from the following train is $s(t)$, and $x(t)$ is the signal received by the preceding train. Hence, $x(t) = s(t)+m(t)$, where $m(t)$ is the channel noise. The signal transferred from the preceding train and received by the following train is $y(t)$.

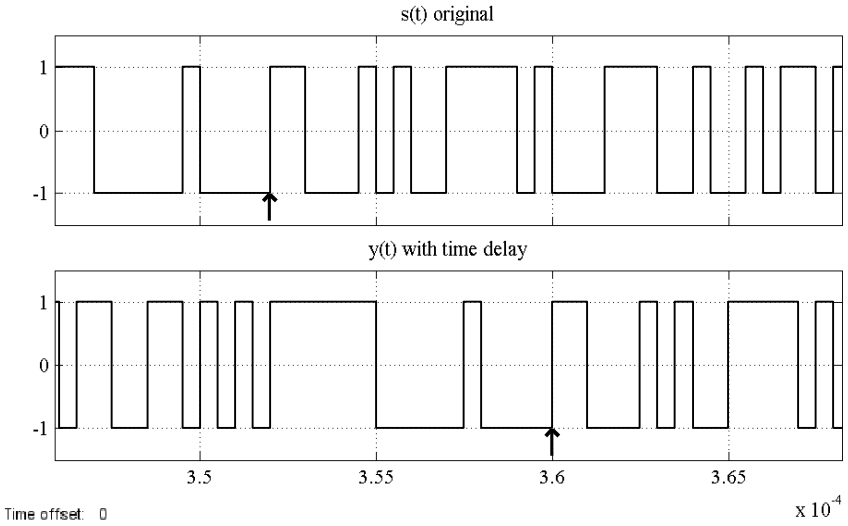


Figure 4.12: Original and with time delay PRN sequences

Taking the time offset in Fig. 4.12 as a case study, the frequency of the original PRN sequence $s(t)$ is 2 MHz, the time delay between $y(t)$ and $s(t)$ is $\tau_0 = 16T_c$, $T_c = 5 \times 10^{-7}s$. Based on the correlation definition, when $\tau = \tau_0$, $R_{sy}\left(\tau-\tau_0\right)$ has a peak point, and the corresponding peak point locates at the $16th$ code element.

As the length of the PRN sequence is $N_c = 63$, the peak points interval is 63 code elements. Fig. 4.13 shows the correlation calculation results of $s(t)$ and $y(t)$, a maximally sharp correlation peak appears at the $16th$ code element. Hence, the minor time delay can be calculated by finding out the corresponding peak point. The distance between two trains is 1200 m.

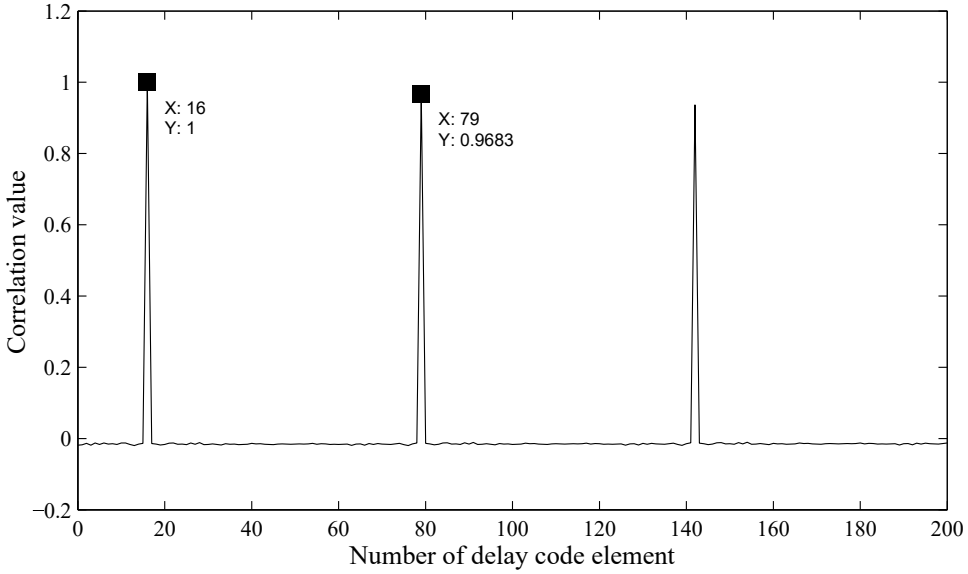


Figure 4.13: $R_{sy}(\tau)$ Correlation calculation

TOA distance estimation flow chart is shown in Fig. 4.14. $s(t)$ is the original signal transferred from the following train, $y(t)$ is signal resent from the preceding train and received by the following train, $m(t)$ and $n(t)$ are the channel noises. The following train does the correlation calculation between the original signal $s(t)$ and the received signal $y(t)$, the τ_0 can be obtained. The principle of calculating τ_0 is introduced in reference [34].

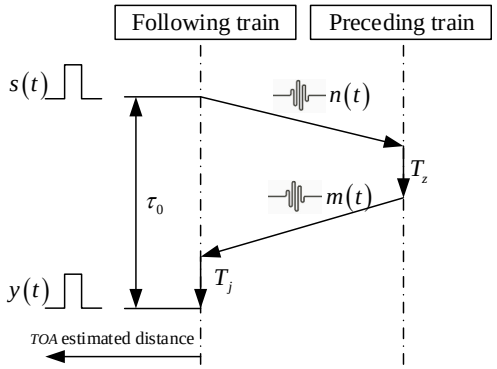


Figure 4.14: TOA distance estimation flow chart

Note that the T has to be modified before actual application, since

$$T = \tau_0 - T_z - T_j \tag{4.20}$$

where τ_0 is total time consumption; T_z and T_j are the time consumed by hardware processing

and calculation, respectively. T_z is a constant value which can be measured in the hardware design. T_j is related to τ_0 and the solution algorithm applied in actual hardwares.

In engineering, the maximum detection distance l_{max} of this method can be estimated by the equation (4.21) [53] .

$$l_{\max} = \left[\frac{P_t G^2 \lambda^2 \sigma}{(4\pi)^3 S_{\min}} \right]^{\frac{1}{4}} m \tag{4.21}$$

where P_t is the transmit power of transceiver unit; G is the antenna gain; λ is the carrier wave frequency; σ is the effective reflective; $S_{\min}$ is the minimum detectable signal.

To implement the system function, a basic block diagram of TTDMS is illustrated in Fig. 4.5. The hardware design will not be discussed here. The PRN generator is responsible for the generation of a PRN code, and the properties of the PRN code control the maximum detection distance and minimum resolution, which will be discussed in section 4.4.2.

Since PRN is often generated from cyclic codes (m-sequences, gold sequences), the resulting correlation graph (Fig. 4.13) shows different peaks. As such, ambiguity may appear. The maximum unambiguous detection of distance is limited by the length of PRN, as shown in equation (4.22). The minimum resolution (P_r) is affected by the code rate of PRN, as shown in equation (4.23).

$$L_u = \frac{N_c \cdot C \cdot T_c}{2} m \tag{4.22}$$

$$P_r = \frac{C}{2 \cdot R_c} = \frac{C \cdot T_c}{2} m \tag{4.23}$$

where L_u is the maximum unambiguity detection distance, N_c is the periodic length of PRN, R_c is the code rate of PRN.

Different carrier waves are applied in different situations. The proper combination of the carrier wave and the antennas can provide a better propagation, and a larger detection distance can be achieved.

4.5 Performance improvement by MA+

The safety improvement by MA+ is analyzed, a procedure is introduced, and the results indicate that the new proposal is flexible and scalable enough for the improvement of railway safety.

4.5.1 Train safe distance interval protected by ATP on-board in ETCS-2

In ETCS-2, the ATP on-board protects the train operation based on the speed-distance monitoring curve, as shown in Fig. 4.15 (a) [54]. First, according to the route and occupation condition of the track section, the RBC generates and sends MA to the ATP. Second, the ATP calculates the permitted speed for every train location according to four categories of data, which are MA, temporary speed restriction, track data, and train data. Third, the ATP compares the actual speed with the permitted speed, if the actual speed exceeds the limitation of brake intervention, the ATP will generate a brake command to reduce the train speed.

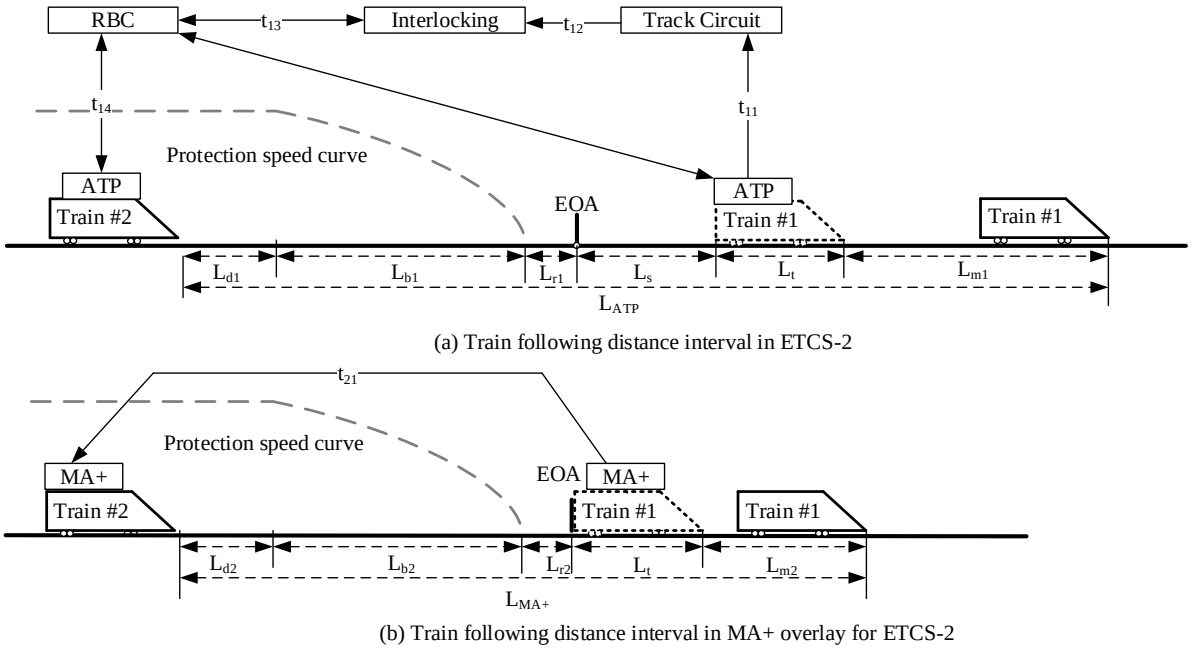


Figure 4.15: Train safe distance interval calculation by ATP and MA+ overlay equipment respectively

In ETCS-2, the End Of Authority (EOA) of the following train locates at the entry point of the block section, which is occupied by the preceding train. Thus, the safe distance interval guaranteed by ATP is given by equation (4.25):

$$L_{EOA} = L_{d1} + L_{b1} + L_{r1} \tag{4.24}$$

$$L_{ATP} = L_{d1} + L_{b1} + L_{r1} + L_s + L_t + L_{m1} \tag{4.25}$$

where

L_{ATP}: safe distance interval when protected by ATP (m);

L_{d1}: additional distance for ATP equipment delay (m), which can be calculated as

$$L_{d1} = v * t_{reaction1} \tag{4.26}$$

where $t_{reaction1}$ represents the time delay (s) for ATP equipment to react to EOA, v represents the train speed(m/s).

L_{b1}: brake distance calculated by ATP (m), which is a function of the train speed, train performance, and line parameters;

L_{r1}: reserved safety distance (m);

L_s: distance from EOA to the rear end of the preceding train in ETCS-2 (m), which can be the length of a block section in the worst case;

L_t: length of the train (m);

L_{m1}: position changing of the preceding train, which relates to the train speed and specific time delay. It is given by equation (4.27):

$$L_{m1} = v * (t_{11} + t_{tc} + t_{12} + t_{IL} + t_{13} + t_{rbc} + t_{14}) \tag{4.27}$$

where t_{11}, t_{12}, t_{13}, t_{14} represent the time delay (s) for the track circuit to detect the train position, message transmission from track circuit to interlocking, from interlocking to RBC, and from RBC to ATP, respectively; t_{tc}, t_{IL}, t_{rbc} represent the processing time (s) of track circuit system, interlocking, and RBC respectively.

4.5.2 Train safe distance interval protected by MA+ overlay system

The MA+ equipment also protects the train operation with a speed-distance curve. The safe distance interval calculation method provided by MA+ is similar with ETCS-2 as shown in Fig. 4.15 (b). In the following train, the MA+ equipment gets an EOA independently based on the safe rear end of the preceding train. This EOA is applied to calculates the permitted speed and protects the train operation in the MA+.

Hence, the safe distance interval guaranteed by L_{MA+} is given by equation (4.28):

$$L_{MA+} = L_{d2} + L_{b2} + L_{r2} + L_t + L_{m2} \tag{4.28}$$

where

L_{MA+}: safe distance interval when protected by MA+ (m);

L_{d2}: additional distance for MA+ equipment delay (m), which can be calculated as:

$$L_{d2} = v * t_{reaction2} \tag{4.29}$$

where $t_{reaction2}$ represents the time delay (s) for the MA+ equipment to react to the EOA.

L_{b2}: brake distance calculated by MA+ equipment (m), which is the same with L_{b1};

L_{r2}: reserved safety distance by the MA+ system (m);

L_t: length of the train (m);

L_{m2}: position changing of the preceding train (m). L_{m2} can be calculated as equation (4.30):

$$L_{m2} = v * (t_{MA+} + t_{21}) \tag{4.30}$$

where t_{MA+} is the time delay for the MA+ module to locate itself and send out its position (s), t_{21} is the time delay for message transmission from the preceding train to the following train (s).

4.5.3 Comparison of the train safe distance interval

Based on the analysis in section 4.5.1 and 4.5.2, we can compare all the components and parameters to find out the performance improvements through applying MA+. It is assumed

that, the train length, speed, reaction time, and the line parameters in both proposals are the same or nearly equal respectively. Hence, the additional distance for equipment delay, brake distance, and reserved safety distance are nearly equal, respectively, which means:

$$\begin{aligned} L_{d1} &\approx L_{d2} \\ L_{b1} &\approx L_{b2} \\ L_{r1} &\approx L_{r2} \end{aligned} \tag{4.31}$$

Furthermore, by comparing equations (4.25), (4.28), we can get the main performance improvement:

- In the MA+ proposal there is no L_s, as the EOA of the following train is the safe rear end of the preceding train's end. However, the EOA is the block section entry in ETCS-2;

- $L_{m2} < L_{m1}$, as the time delay for the MA+ proposal is much less than it in ETCS-2. We can reasonably assume that the time delay for message transmission via wireless channels in both proposal is the same; the processing time for MA+ is no more than Track Circuit module, which means $t_{21} < t_{13}$ and $t_{MA+} < t_{tc}$. Hence, this deduction based on equations (4.27) and (4.30) is obtained.

Based on these two differences, we can get $L_{MA+} < L_{ATP}$, which indicates that there is a big decrease of the safe distance interval from the ETCS-2 to MA+ proposal. In another word, the MA+ equipment can brake later than ETCS-2 under the premise of keeping trains' safe separation. It is clear to find out that the MA+ proposal can assure the system safety without disturbing the normal protection function in ETCS-2, which is of great importance for a backup equipment.

To prove the effectiveness of the MA+, we also calculated the safe distance interval in ETCS-3. Although it is still in the conceptual stage, ETCS-3 is an upgrade level of ETCS-2. Hence, it is useful to make the comparison between the MA+ proposal and ETCS-3. In ETCS-3, the RBC receives the safe location of the preceding train, which is generated by the ATP, and sends it to the following train as the EOA, and the train is operated based on the moving block theory. Hence, the time in both ATP and RBC should be taken into consideration. The position change in ETCS-3 L_{m3} can be calculated as:

$$L_{m3} = v * (t_{atp} + t_{15} + t_{rbc} + t_{11}) \tag{4.32}$$

where t_{atp} is the processing time for ATP in the preceding train to locate itself and send its position to RBC; t_{15} and t_{11} are the time consumed by the message transition from the preceding train to RBC and from RBC to the following train, respectively; t_{rbc} represents the RBC processing time, including the communication cycle between RBC and On-Board ATP.

To see it intuitively, we simulate and obtain the safe distances under different train speeds in three different solutions. The value of the parameters we use in the simulation are shown as table 4.4, most of the parameters are the typical value in the Chinese High-Speed Railway [55].

Table 4.4: Parameters used in the simulation

Time Parameters	$t_{reaction1}$	t_{11}	t_{12}	t_{13}	t_{14}	t_{tc}	t_{IL}	t_{rbc}	$t_{reaction2}$	t_{21}	t_{MA+}	t_{atp}	t_{15}
Value (s)	1	0.5	0.05	0.2	0.5	2	0.2	6	1	0.1	0.2	6	0.5
Length Parameters	L_{r1}	L_s	L_t	L_{r2}									
Value (m)	200	2000	416	200									

As shown in Fig. 4.16, the safe distance interval in MA+ is much less than it in ETCS-2 and ETCS-3 at any speed. As has analyzed before, the main reasons for the safe distance interval decrease are the elimination of the fixed block length in ETCS-2, and the time saved from direct train-to-train real time communication compared with a lot of processing nodes and time delay in ETCS-2 and ETCS-3. ETCS-2 relies on the RBC to send the EOA to ATP based on fixed block principle, so the safe distance interval is always much longer than it in MA+.

The results also indicate that the safe distance interval in ETCS-3 is much shorter than it is in ETCS-2 at low speed, but the gap decreases quickly as the speed grows. There are two reasons for this situation: i) the safe distance interval in ETCS-2 includes a block section length L_s, which makes the distance interval longer than it in ETCS-3; ii) position changing of the preceding train L_{m1} in ETCS-2 is shorter than L_{m3} in ETCS-3, as the sum of all the message transmission delay time in equation (4.27) is less than it in equation (4.32). Hence, when the train speed is getting lower, the block section length weights higher and the gap between these two levels is bigger; as the train speed grows, the gap becomes smaller.

This is also one of the main reasons why the adjacent MA+ module should communicate with each other much more frequently (every process cycle time, 0.2 s) than ATP with RBC in ETCS-2 or ETCS-3 (every communication cycle time, 6 s), so MA+ is better than ETCS-3 as we can see in Fig. 4.16. As a result, the MA+ proposal can work well as a backup module

for ETCS-2 and even for ETCS-3, which can brake later when the main system fails and still be able to ensure system safety.

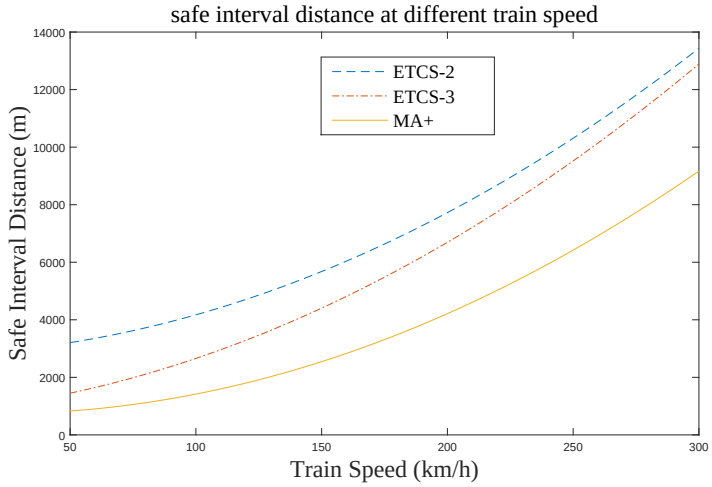


Figure 4.16: Safe distance interval in different speed

4.6 Summary

Except for the train-centric wireless communication architecture, other hardware and information are shared between the new movement authority system and existing onboard equipment. Integrating with the existing onboard equipment, this system has two strategies to carry out the train distance interval calculation. Importantly, TTDMS solution is possible in emergency situations. For example, the onboard equipment outputs erroneous commands or the connection between train and waysides equipment is abnormal.

Additionally, it was crucial to confirm the safety of hardware design, and to guarantee the system correctness. Hence, the reliability of a prototype TTDMS was estimated. However, to compare with the actual data, a new TTDMS prototype machine can be developed, which has a further detection distance, and the experimental data can be collected in real train operation scenarios. What is more, it can permit the consideration of both simulated and analyzed real data.

For the system safety analysis, an approach was proposed to calculate the safe distance interval and dangerous side failure rate of the new proposal compared with the traditional ETCS-2 system. The results indicated that the safety increase ration is greater than 1.5, especially this new system is more suitable for the lines with few trains and more wayside equipment.

5 System modeling and validation by means of Petri nets

Validation, verification, and evaluation are necessary processes to assure the safety and functionality of a system before its implementation in practice. As suggested in EN 50128 [56], in the area of railway application, the techniques of formal methods are suitable to do the system requirements specification, design, implementation, as well as modeling.

In this chapter, we introduce the formal methods that can be applied to do system evaluation and verification, and the benefits of using CPNs in this thesis. Afterwards, basic definitions of CPNs and tools for the application of Petri nets are presented. Finally, the methods that can refer to the qualitative items and quantitative values in the `CPN Tools` are illustrated.

5.1 Evaluation methodology

A system has to be evaluated during its life cycle. The evaluation results provide essential information for system design and improvement. System performance represents the effectiveness (safety or reliability) of a system. The performance verification helps engineers to have a better understanding of the system before its application in practice. Model and simulation can obtain the information about how a system reacts, without actually observing the data from experiments in practice. With proper parameterization, the model can represent fundamental aspects of the system, and obtain data that is either hard or costly to replicate in the real world.

A parameterized analysis is possible in the simulation process. The system accuracy indicates how well the system operates. It can analyze the system according to reasonable parameterizations. Product data is the basis of information technologies in development and manufacturing processes. The data arises in the complete life cycle of a product and are required in all departments of production enterprises.

The evaluation methodology helps one better understand the system [57]. There are four main steps in the evaluation methodology:

- Determine the parameters of the evaluation.
- Determine the methods used for the evaluation.
- Set requirements and collect data.
- Analyze and make conclusions.

There are a number of operational and technical questions that must be explored as a basis for evaluating these capabilities. As an example, fault tree is necessary to evaluate the whole system reliability, but insufficient to provide the performance details. See Table 5.1. A number of evaluation methods were examined for their applicability to answering relevant questions. These methods are scoped as follows:

Mathematical calculation: Based on scientific formula derivation to estimate the system behaviors, and to include component evaluation of a technology in the operational setting under controlled environmental conditions.

Fault Tree and Hazard Analysis: Techniques used to ensure that a system behaves as needed when parts fail.

Model checking: Given a model of a system function, exhaustively and automatically check whether this model meets a given functional specification

Monte Carlo Simulation: combines kind of models for each technology to capture the statistical distributions. Permits extensive testing runs that may not be practical under real operating conditions; used to scope the performance bounds of the technology.

Laboratory Simulation: provides operating capabilities in a simulated environment, to supplement field testing and as a precursor to operational evaluations; frequently done in parallel to advance the understanding of the technology.

Operational Evaluation: provides a full range of testing to prove that a technology fulfills a specification or standard. Includes limited deployment in the operational environment.

Performance Monitoring: adds to the technology experience data, by evaluating it in use, under standard operating conditions.

Table 5.1 maps evaluation methods to some of the potential research and development questions that exist in the aviation community. No one evaluation method can answer all the

questions. Thus a well constructed set of tests must be developed to sufficiently support evaluation, verification, and certification of collision avoidance capabilities.

Table 5.1: Evaluation methods, and the questions that can be addressed

	Mathematical calculation	Fault Tree and Hazard Analysis	Model checking	Monte Carlo Simulation	Laboratory Simulation	Operational Evaluation	Performance Monitoring
Are the system structure correct?			√				
Can a specific reaction be implemented?			√	√	√		
Is the model correct?			√				
Does the system function together?		√	√	√	√	√	√
Does system provide safe separation from trains?	√		√	√	√	√	√
Does collision avoidance algorithms react in an acceptable way for other trains?	√			√	√	√	√
What are the limits of the system? Conditions? Range? Reliability?		√	√	√	√	√	√
How well the system performance? i.e., time delay	√		√	√	√	√	√
What is the overall system performance?		√		√	√	√	√
What does the system range?	√				√	√	√

5.2 Formal methods for system evaluation and verification

The train protection system is a safety-critical system. The development of train protection system has been to reduce the reliance upon and the responsibility placed on train drivers to maintain safe separation between trains. It has been done by providing different improved signaling systems. The issues related to system safety are of primary importance [58]. To assure safety and functionality of each system before its application, an efficient development and analysis procedure that can perform verification and validation analysis is essential.

Various methods can be applied to evaluate the reliability performance of the safety-related system. For instance, fault tree analysis [59], Bayesian network [60], and Markov Analysis

[61]. However, fault tree analysis is based on the assumption that all items obey Bernoulli distribution, which is not suitable to represent practical systems [62]. The Bayesian network cannot deal with system functional safety evaluation. Markov chain does not allow the use of non-exponential statistical. Importantly, real systems require a higher formalization level to describe.

As the high level Petri nets, Colored Petri Nets (CPNs) describe system actions and states by using definitions of different color sets and data manipulation. The aim of this thesis is to validate, verify and evaluate the target system's properties by means of CPNs. Hence, an approach is proposed in this thesis for the evaluation and verification of system description and development. Importantly, different from most analysis methodologies for systems that only perform simulations to verify the systems' performance or the correctness of system functions, this approach provides both functional safety evaluation and simulation-based verification. CPNs can carry out these requirements.

There are several motivations using CPNs:

- CPNs have a discrete-event modeling language, which combines basic Petri nets with the functional language Standard ML (SML);
- CPNs describe system actions and states using the definition of different color sets and data manipulation;
- and CPNs support modeling, formal analysis, and simulation [62].

For these purposes, CPN method is chosen to implement the formalization. What is more, Petri nets have been applied for the modeling and evaluation of safety-critical systems [63–66], such as a multitude of successful research projects implemented by Institut für Verkehrssicherheit und Automatisierungstechnik (iVA), Technische Universität Braunschweig, Germany. These are motivations using CPNs for a model based reliability analysis and evaluation.

More importantly, taking into account that railway-related systems are highly complex systems in aspects of functionality as well as dependability [67], a procedure is proposed to analyze the system safety and security during the TTDMS development. Formal methods, e.g. Petri nets, are suggested in EN 50128 for railway applications' system design as well as modeling [56].

In the railway domain, UML, the B method, Petri nets and other varieties of modeling languages have been applied to describe railway application systems [63]. UML is usually used to model the system and software functionality, but is not suitable for structural analysis

and formal proof. The B method is mainly used for function modeling and proof as well the source code generation. CPNs, as one of such formal methods, permit the description of a system in development phases as diverse as system requirements specification, design, evaluation, and coding [34].

CPNs are usually applied in the modeling of systems and the verification of software functionalities [68]. Additionally, the reachability graph is used for the actual software development down to source code generation. Hence, CPNs are used as the formal method to provide a framework of the system, and a proper software needs to model the system structure and analyze the system performance.

The systems development lifecycle can be divided into two main sections, as shown in Fig. 5.1 [56]. CPNs can be applied into the whole period to implement the system description, evaluation, and verification. What is more, the CPNs can also be applied to carry out the system performance evaluation after the system implementation, and more details can be found in our previous publication [34].

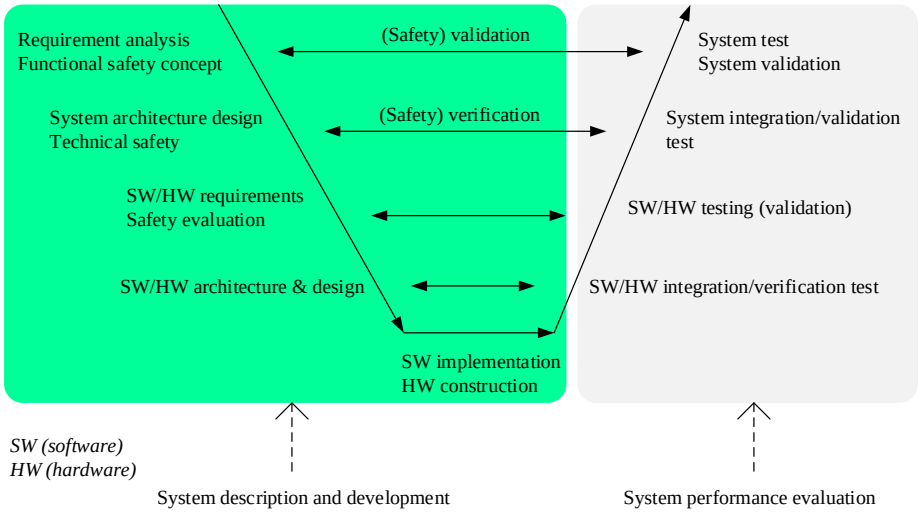


Figure 5.1: Systems engineering and verification

5.3 Basic definitions of CPNs

CPNs language is a discrete-event modeling language and combines basic Petri nets with a functional language, i.e. Standard ML (SML). CPNs provide different color sets, which are suitable for the parameterization process. In CPN models, each place represents a

system state. Transitions between places can interpret as activities. For a model net $\mathcal{N}et = (\Sigma, P, T, A, N, C, G, E, I)$ satisfies the requirements below [69]:

- Σ is a finite set of **color sets**. It represents different resources.
- P is a finite set of **place**. It indicates states.
- T is a finite set of **transitions**. System action can be treated as transition in CPN.
- A is a finite set of **arcs**.
- N is a **node** function, it is defined from A into $(P \times T) \cup (T \times P)$.
- C is a **color** function. It is defined from P into Σ.
- G is a **guard** function. It assigns a guard to each **transition** t such that $Type[G(t)] = Bool$.
- E is the **arc expression** function. For a given **place** $p \in P$, $E(p, t)$ represents the arc expression on the input arc from p to t.
- I is an initialization function. It is defined from P into closed expressions such that: $\forall p \in P : [Type\,(I\,(p)) = C(p)_{MS}]$. Here, MS refers to multiset.

Additionally, the global state of a given $\mathcal{N}et$ at a given moment can be depicted as situation, i.e. the current marking of the net is M, and all the markings in another labeled order combine an occurrence graph OG [70]. M_0 is the initial marking. The set of markings reachable from a marking M is denoted as $\mathcal{R}(M)$.

Fundamental terms and concepts of a system are resources, rules, input and output, intermediate data, and processes. Formal validation should cover as many of these as possible. For instance, rules and processes are related to system behavior; intermediate data and input/output can be grouped as parameterization. The relationship between system structure and Petri net description is shown in Fig. 5.2. Hence, the corresponding relationship can be used to assist the system development. A system includes two main properties: states and functions. States show the system's situations. Functions are in charge of changing the system states by means of events. Different system states trigger respective events, and the updates of events trigger respective states by means of events. After the basic properties are clear, more system structure details should be considered. CPNs represent system states and events as places and transitions in the model, respectively.

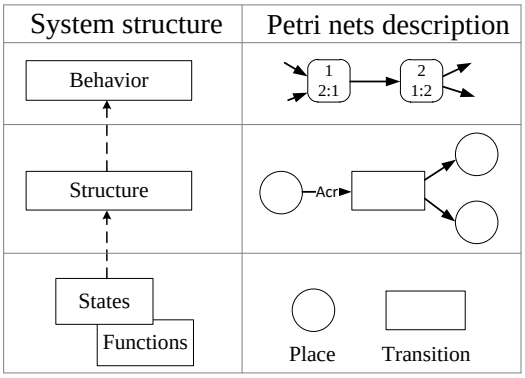


Figure 5.2: Relationship between system structure and Petri net description

5.4 Tools

There are different kinds of computer-aided tools for the application of Petri nets. They support different Petri nets and features. Normally, four basic kinds of Petri nets are token into consideration: low-level Petri nets, high-level Petri nets, stochastic Petri nets, and timed Petri nets. Based on the usage of this thesis, some tools are shown in Table 5.2, and the features of these tools are defined as follows [63]:

- `Graphical Editor` supports editing of nets in a graphical representation.
- `Token Game Animation` supports simulation with animation of the flow of tokens.
- `Simulation` supports simulation without graphics to allow maximum simulation performance.
- `State Spaces` supports the generation of markings (also know as reachability graphs/trees and occurrence graphs).
- `Performance Analysis` permits performance analysis such as simulation with time, Markovian chains, Monte Carlo simulation, and so on.

Based on the requirements of this thesis, the simulation, state space, and performance analysis are the basic required features. Hence, the software CPN Tools is applied in modeling the system structure and analyzing its performance. CPN Tools is a software for editing, simulating, and analyzing CPN models; various stochastic scenarios can be simulated in the model. It can be set to follow different stochastic distributions to monitor real situations. What is more, in some case studies, the $\pi-$Tool is applied for the stochastic Petri nets analysis. In this paper, the high-level Petri nets analysis is not carried out by $\pi-$Tool. As

Table 5.2: Petri net tools and their features

PN and Features supported	Tools					
	CPN Tools	$\pi-$Tool	Poseidon	TimeNET	INA	GreatSPN
Low-level Petri nets		X	X	X	X	
High-level Petri nets	X	X		X	X	X
Stochastic Petri nets		X		X		X
Timed Petri nets	X	X	X	X		X
Graphical Editor	X	X	X	X		X
Token Game Animation	X	X	X	X		X
Simulation	X	X		X		X
State Spaces	X	X	X		X	X
Performance Analysis	X	X		X	X	X

the $\pi-$Tool is more suitable to do the simulation of stochastic analysis, which not take the high-level Petri nets into consideration.

5.5 Items and data value

5.5.1 Qualitative items

A state space is a occurrence graph of CPN models. In this graph, a node and an arc represent each reachable marking and occurring binding, respectively [69]. M_i used to represent the marking i in the state space. The variables of a transition t denotes $Var(t)$. All bindings for a transition t is denoted $B(t)$. A binding element is a pair (t, b). The system state will be modified through firing specific transitions. In a marking M_i, when an enabled binding element (t, b) occurs, it will change the marking M_i to another marking M_{i+1}, defined by:

$$\begin{aligned} &\forall p \in P : M_{i+1}(p) = \\ &M_i(p) - \sum_{(t,b)\in Y} E(p,t) < b > + \sum_{(t,b)\in Y} E(t,p) < b > \end{aligned} \tag{5.1}$$

where $M_i(p)$ is the number of $M_i(p) - \sum_{(t,b)\in Y} E(p,t) < b >$ removes tokens from M_i while $\sum_{(t,b)\in Y} E(t,p) < b >$ added tokens to M_{i+1} . Moreover, M_{i+1} is directly reachable from M_i by the occurrence of the step Y, which denote: $M_i[Y > M_{i+1}$.

The analysis of the state space is carried out by checking queries with Computational Tree Logic (ASK-CTL) [71]. The ASK-CTL logic and the model checker are implemented in SML [72]. To denote place instance, transition instance, and markings, the following SML structures are available, respectively.

```
con PI.<PageName>'<PlaceName>
```

```
con TI.<PageName>'<TranName>
```

```
fun Mark.<PageName>'<PlaceName>
```

SearchNodes is the basic function that traverses the state space. The function can be described as Fig. 5.3 in pseudo-code, more details are available in publication [72].

- **Area** specifies the searching area, data type **Node** list.
- **Pred** evaluates whether the **Node** can take part in further analysis or not, data type `Node -> bool`.
- **Limit** specifies searching terminal requirement, data type `int`.
- **Eval** maps each **Node** into a value, data type `Node -> 'a` [2].
- **Start** is a constant, data type `'b`.
- **Comb** is a function can map **Eval(n)** and **Result** into `'b`, data type `'a*'b -> 'b`.

SearchNodes(Area,Pred,Limit,Eval,Start,Comb)

```
SearchNodes(Area,Pred,Limit,Eval,Start,Comb)
Begin
    Result:=Start; Found:=0
    for all n in Area do
        if Pred(n) then
            Result:=Comb(Eval(n),Result)
            Found:=Found+1
            if Found=Limit then
                break
            end
        end
    end
    return Result
end
```

Figure 5.3: Pascal-style pseudo-code of basic traverse method

[2] `a`, `b` are arbitrary data type

The exact SML call for SearchNodes looks as follows:

SearchNodes$(EntireGraph, fnn => (length(OutArcs(n)) = 0), 10, fnn => n, [\,], op ::)$

PredNodes and EvalNodes are abbreviated forms of SearchNodes, functions details shown as follows:

PredNodes$(area, pred, limit) : Nodelist = SearchNodes(area, pred, limit, id, [\,],, op ::)$

EvalNodes$(area, eval) :' blist = SearchNodes(area, fn\_ => true, NoLimit, eval, [\,],, op ::)$

PredNodes and **EvalNodes** are abbreviated forms of **SearchNodes**. Functions details shown as follows:

PredNodes$(area, pred, limit) : Nodelist =$ **SearchNodes**$(area, pred, limit, id, [], op ::)$
EvalNodes$(area, eval) :' blist =$ **SearchNodes**$(area, fn\_ => true, NoLimit, eval, [], op ::)$

PredNodes$(area, pred, limit) : Nodelist= SearchNodes(area, pred, limit, id, [\,], op ::)$

EvalNodes $(area, eval)\ : 'b\ list = SearchNodes(area, fn\_ => true, NoLimit, eval, [\,], op ::)$

Similarly, functions **SearchArcs**, **PredArcs** and **EvalArcs** traverse the **arcs** of the state space. Boundedness properties are used to check how many and which tokens on a particular place instance. There are four query functions for the boundedness properties:

```
fun UpperInterger (Node->'a ms)->int

fun LowerInterger (Node->'a ms)->int

fun UpperMultiSet (Node->'a ms)->'a ms

fun LowerMultiSet (Node->'a ms)->'a ms
```

As shown in following Fig. 5.4, there are three numbers in the rounded rectangle. The number in the first line is the marking index number, for instance, "1" means the marking 1. The second line contains two numbers, taking marking 1 as an example, the left number "2" indicates only one marking can transfer to marking 2, the right number "1" means marking 1 has one successor.

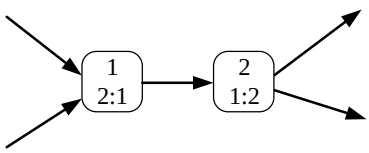


Figure 5.4: Partial state space

5.5.2 Quantitative values

When simulating or executing a CPN model, it is necessary to extract information from the markings and binding elements, and then to do the analysis. Data collector monitors are used to extract numerical data during simulations. The numerical data is used when Calculating statistics, and the data may be saved in data collector log files. A data collector can calculate either timed or untimed statistics. Data collector monitors are used for performance analysis of CPNs. There are four kinds of data collector monitors: marking size monitor, list length monitor, count transition occurrences monitor, and generic data collector.

Function types for the accessible functions:

- init: markings -> <numerical type> option initialization function
- pred: subnet -> bool predicate function
- obs: subnet -> <numerical type> observation function
- stop: markings -> <numerical type> option stop function

Normally, the observation function of the generic data collector is mainly applied in the practical simulation. This function examines the monitored nodes, and returns a numerical value. For more information on data collector monitoring functions, please refer to the publication of [72].

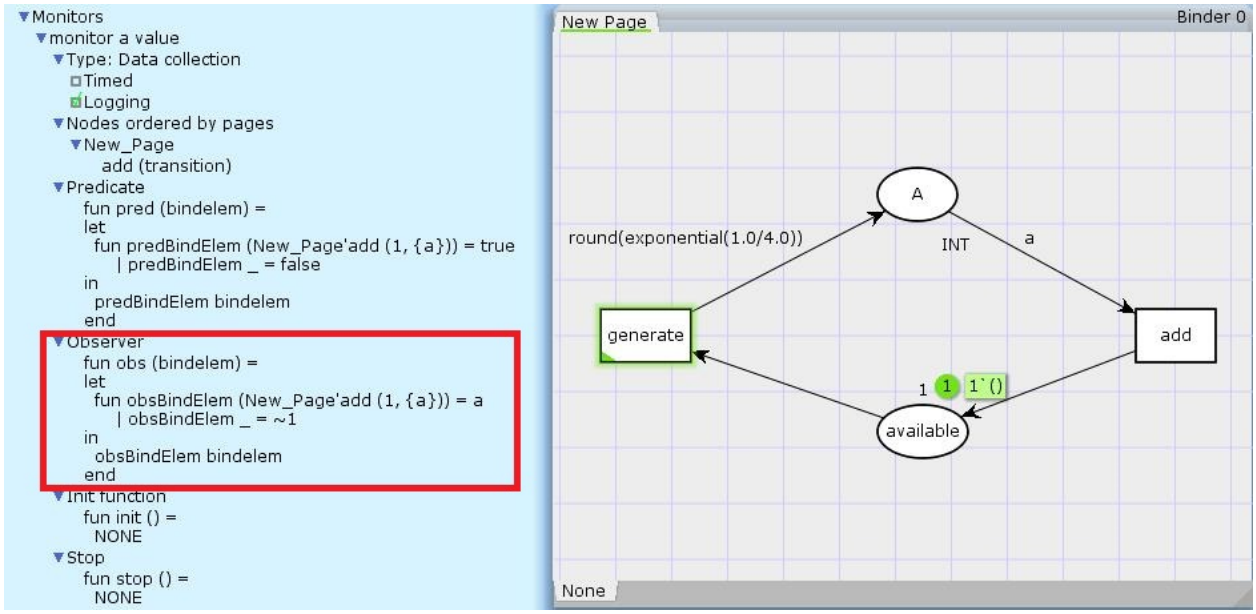


Figure 5.5: A net consists of place `A` and its surrounding transitions on page `New Page`

Taking the following model as a case study, as shown in Fig. 5.5. The transition `generate` outputs values which follow an exponential distribution with the mean value 1.0/4.0. To collect the data value, the monitor named as "monitor a value" is associated with the transition `add`. If the predicate function is invoked after the `add` occurs, then the observation function will return true. The observation function will be invoked when the predicate function for the monitor returns true, and it will return the value of bind element `a`. The data values are collected during a simulation and should be saved in a data collector log file.

5.6 Problem description and definitions

A variety of modeling languages and methods can be used to describe railway application systems [63]. All these approaches can be described by the BMW-concept [73], which is developed by the Institute for Traffic Safety and Automation Engineering (iVA), Technische Universität Braunschweig:

- B is Beschreibungsmittel in German, it denotes the means of description;
- M is Methoden in German, it indicates the methods for design, structuring, modeling and verification;
- W is Werkzeuge in German, it represents the tools for system engineering to support methods and description.

The means of description consist of informal and formal ones. An example of the informal description is the human language, which has an ambiguous attribution. In order to describe a system more specifically, the informal description is translated into computerized formalization.

The development of safety-critical systems leads to complex control structures. In this chapter, a common approach for the development of a new system is divided into four stages, as shown in Fig 5.6:

- conceptual level,
- construction level,
- formal level,
- and technical level.

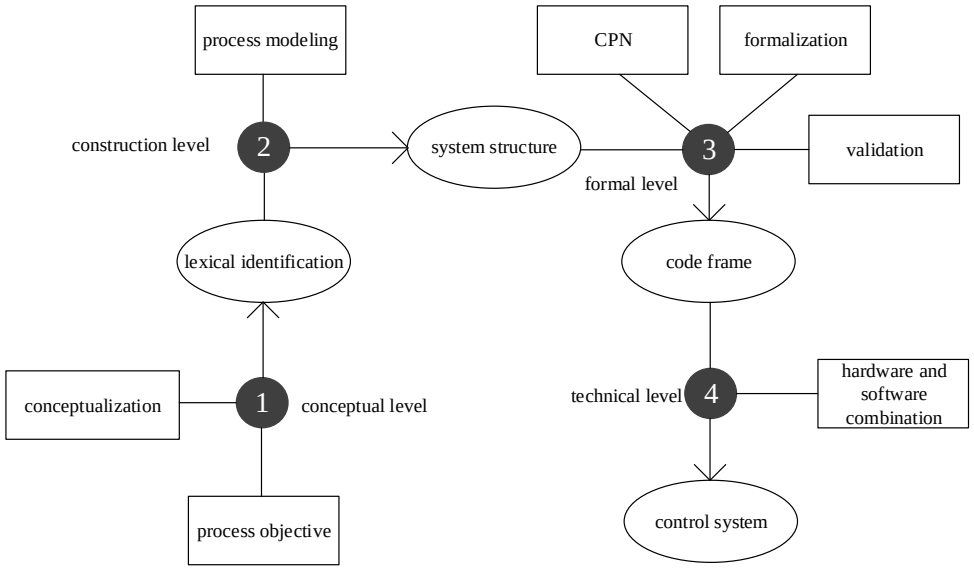


Figure 5.6: A common approach for system development

In this chart, square boxes represent available resources or methods, and ellipses represent outputs. The arrows indicate the development procedure.

In step 1, which has been elaborated on in the aforementioned chapters, a system starts from conceptualization and objective. The system's requirements are illustrated in this step. After the identification of need is carried out, the system's requirements and functions are presented. However, the connections between possible scenarios cannot be specified unambiguously when an operation process is described using natural language. Hence, step 2 advances this approach.

In step 2, before the engineers transfer the concept and objective into a system structure with the help of informal process modeling, it is necessary to have a basic understanding of the system principles. This step is carried out in chapter 4. Normally, informal formulations are like charts or blocks, such as the block diagram in section 4.1. After introducing the system principle, we illustrate application scenarios of TTDMS in a chart flow, and the system block diagram is also presented. Because of the informal description of the system structure, no validation of system behaviors is possible. With the assistance of the unambiguous definition of formal methods, a system can be presented in a clear way. Hence, the procedure can be furthered by using formal methods.

In step 3, the system structure is formalized into a CPN model, which is implemented in section 6.1. CPN-modeling is a suitable means of description for representing system behaviors, and validation can also be implemented by using the state space analysis [33].

Modeling the system structure with CPNs permits derivation of an occurrence graph. The chart involves all procedural sequences of the operational process. Based on the graph, the system functionality and behaviors can be validated. After the model structure's safety is validated, we present a universal approach to generate code frameworks from the occurrence graph.

In step 4, we assemble an actual TTDMS using a combination of hardware and software. Sections 6.3 and 7 provide the analysis of TTDMS performance. TTDMS performance is evaluated by implementing in the simulation, mathematical calculations, and model based estimation. Additionally, actual measurement results are presented and analyzed. The results indicate that TTDMS can meet the requirements of daily metro trains. This can be found in chapters 9 and 10.

In this chapter, the new process seeks to perform TTDMS modeling and validation, functional safety and timed performance evaluation using CPN, and estimates the actual system reliability. Here, we combine the system development and formal method conceptualization together. The procedure can be divided into four steps, as shown in Fig. 5.7.

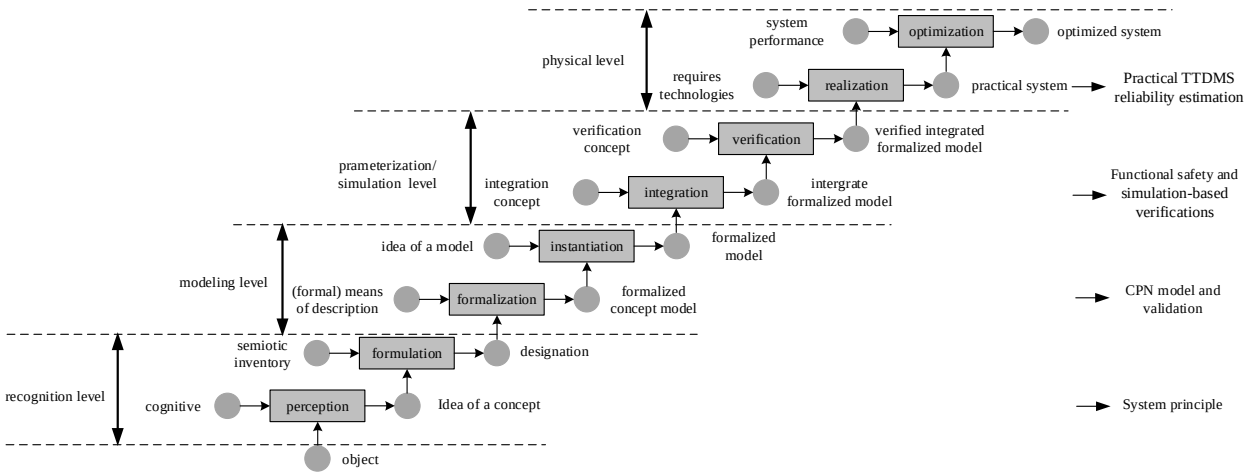


Figure 5.7: System development and formal method conceptualization (adapted from [1])

- Recognition level. The development of increasingly sophisticated automation systems usually starts with a verbal requirement specification [1]. Each new system starts with the activity of human mind, and the mind has to be described and explained using languages. Consequently, other individuals can understand the system.

- Modeling level. Considering all languages have ambiguities, people can rely on formal methods to describe the idea. With the assistance of the unambiguous definition of

formal methods, a system can be presented in a clearer way. People can take advantage of formal models to reflect human ideas and the system behaviors.

- Parameterization/ simulation level. It carries out the analysis of system functional safety and performance. The functional safety determines a system can operate as designed. An actual system operates in real time space, but some time-related performances cannot be expected for a relatively long time. Hence, after the language description is transferred to a formal model, functional safety analysis and simulation based on the model are required. As a result, engineers can rely on the results to update the actual system.
- Physical level. A physical system is available, and the system reliability is estimated.

5.7 Summary

In this chapter, system evaluation methodologies were discussed. The results indicated that a well-constructed set of tests were required to support the evaluation and verification of a system sufficiently. To cover the whole system development life cycle, CPN method was chosen to implement the formalization. After introduced the basic definitions of CPNs, different kinds of computer-aided tools for the application of Petri nets were discussed. Additionally, we briefly introduced the method to refer the items and data value of CPN models, which were used for the further functional and performance analysis. Finally, we introduced a novel approach, and it was divided into four levels and four steps to carry out the system development, functional safety evaluation, and simulation-based verification. The instantiation of this approach is carried out in chapter 6.

6 Formal modeling of TTDMS

In this chapter, we proposed a new process using Colored Petri Nets to verify the TTDMS system functional safety, as well as to evaluate the system performance. Three main contributions are carried out in the chapter: Firstly, this chapter proposes a formalized TTDMS model, and the model correctness is validated using state space analysis and simulation-based verification. Secondly, corresponding checking queries are proposed for the purpose of functional safety verification. Further, the TTDMS performance is evaluated by applying parameters in the formal model. Thirdly, the reliability of a functional prototype TTDMS is estimated. It is found that the procedure can cooperate with the system development, and both formal and simulation-based verifications are performed. Using our process to evaluate and verify a system is easier to read and more reliable compared to executable code and mathematical methods.

6.1 Formal modeling process of TTDMS

Based on the unambiguous definition of formal methods, the TTDMS can be described much clearer and more readable than it is in executable codes. Following the steps in Fig. 5.7, the system will be modeled by using CPN. Based on the system property concept proposed in this section, an instantiation model is built and validated.

6.1.1 Formal modeling and system property

Generally, system operation can be described as transitions between different system states. A series of system operations can be treated as a transition-firing sequence from one state to another. Each system state can be represented as a place in the computerized executable model. The model parameterization can implement the mathematical description and the unit dimension. The description refinement of a system depends on the formalization degree.

The higher the formalization level used to describe the real system, the greater the possibility to formally (mathematically) verify the formalized concept system [1].

To evaluate the system performance before its actual design, a computerized executed model is required. Here the system property concept that proposed by iVA is applied to build the CPN model [74]. Four hierarchical levels of system abstraction are composed and described as follows [75].

- Property. The property represents the abstraction of characteristics. Observable characteristics empirically cause observed properties, and properties deal with abstract concepts.
- Characteristic. The characteristic is the fundamental element of objective recognition and description. Therefore, the characteristic is objectively determinable, and it describes the essential or unique feature of the property. Characteristics are available according to a measurement (continuous characteristics) or counting (discrete characteristics). The characteristics can be quantified, and they can be directly described and identified by digital objects.
- Quantity. The quantities are physically and numerically deduced from characteristics. Quantities are special cases of the general characteristic values. The quantities are determined due to the restriction of related characteristics.
- Value & Units. Every quantity results in one value and its unit. Values result from measurements and calculations. A numerical value including its unit can also be parameterized into system quantities.

6.1.2 CPN model of TTDMS

Since the TTDMS availability and functional safety are the most important parts, the different communication parameters should be considered in the model process. The communication system stability influences the distance measurement accuracy. Many different parameters affect the TTDMS performance. In a normal operation mode, the communication link between two TTDMS is connected, the following failures can occur [76]:

- Fail to establish a connection. To execute a communication, the first step is to establish a communication link, and then exchange data. The time consumed in establishing the connection is uncertain, and a reconnection action is required if it fails.

- Transmission error. Due to various reasons, there is not enough data information available to perform an actual distance calculation during a short period. Importantly, the communication link is still available.
- Disconnection. The channel link between two systems is broken and a reconnection is required. This state normally happens in a poor signal transmission environment.

Availability and functional safety are selected as two fundamental properties of the TTDMS. The availability emphasizes how well a system operates in numerical analysis. The functional safety indicates whether a system runs normally in logical analysis. The system availability can be described with distinct characteristics, such as connection stability and time delay [77]. The characteristics of functional safety indicate that the TTDMS can react and work correctly under specific resources.

Hence, the CPN model and system attribute hierarchy is shown in Fig. 6.1. For the TTDMS, the distance information should be updated in time. The time delay has to be limited to an acceptable level, which will not influence the train operation. Here three main parameters are chosen with in the quantity level to quantify the characteristics of the connection stability: time consumed during connection establishment, transmission error, and disconnection. The token, available markings and path can be used to implement the functional safety analysis. The detailed description of system functional safety is introduced in section 6.2, and the feasibility analysis is presented in section 6.3.

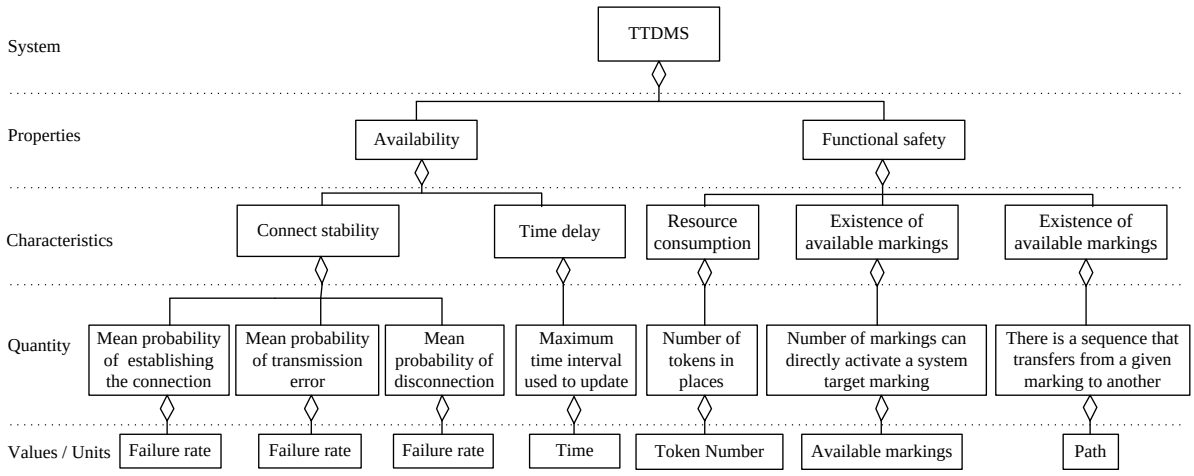


Figure 6.1: Concept model of the TTDMS in the structural property aspect

The CPN model of TTDMS is built as shown in Fig. 6.2, and Table 6.1 shows the corresponding meanings of places and tokens in the model. The red color marked part indicates the abnormal procedures and states. While, the green color part means the correct

process. As shown in the model, the TTDMS starts with 1‘IDLE under the surveillance mode. If it detects other TTDMS, it tries to establish a connection with the transition try to scan. Once the communication link is established successfully, the following system sends the PRN code and attempts to calculate the distance. Transitions `data_loss_1` and `part_transmission_1` represent disconnection and transmission error, respectively.

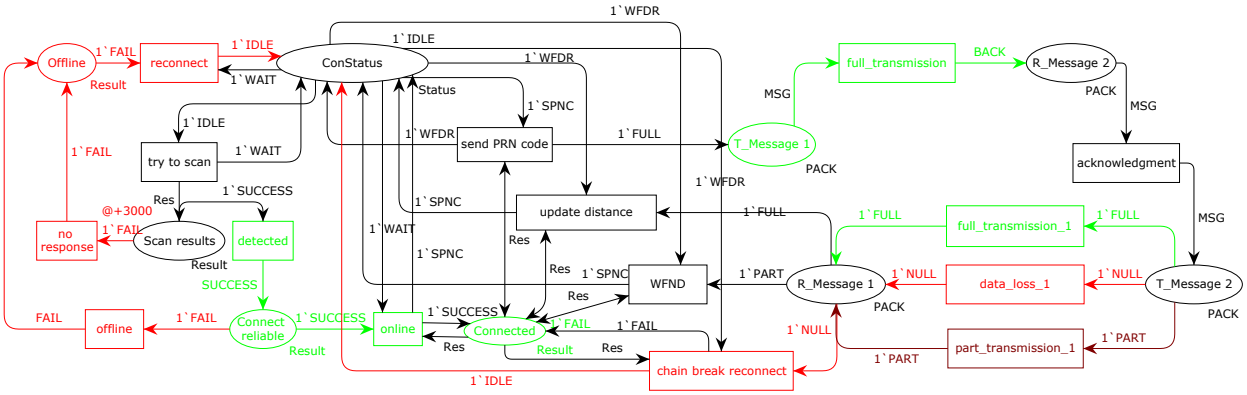


Figure 6.2: The CPN model of the TTDMS

Table 6.1: Meanings of places and tokens in the CPN model (see Fig. 6.2)

CPN model element	Element type	Meaning
ConStatus	Place	Main system situation
Scan results	Place	Detection result
Connect reliable	Place	Connection situation
Offline	Place	Disconnection
Connected	Place	Connection established
R_Message 1	Place	Received message in the following train
T_Message 1	Place	Transmitted message in the following train
R_Message 2	Place	Received message in the detected train
T_Message 2	Place	Transmitted message in the detected train
IDLE	Token	Connection is closed or does not exist
WAIT	Token	Wait for detection results
SPNC	Token	Send PRN code and try to calculate the distance
WFDR	Token	Wait for feedback data from the detected train
FULL	Token	Completed packet
PART	Token	Incomplete packet
NULL	Token	Packet lost

6.2 Model validation and functional safety verification

In safety critical applications, system validation is an essential step that is required before further developments. Normally, model checking is an automatic validation technique that can implement analysis of the system behavior. In this study, we applied the model checking facilities of the *CPN Tools* to investigate requirements of TTDMS. As illustrated in Fig. 5.2, the system behavior results from the combination of a change in state and the execution of functions. Each relevant physical and technical state of a component is called a local state [27]. Ensuring that functions can generate correct states, and that correct states can trigger appropriate functions are the most important requirements in system design.

In section 6.2.1, four validation requirements are proposed. Both the requirements' meanings and methods have been performed. Corresponding checking queries with ASK-CTL are also demonstrated in this section.

After the formalized model is validated, the functional safety is verified using state space analysis in section 6.2.2. The basic components of a functional system are resource, rules, input/output, intermediate data, and process. Formal verification should cover all these elements or as many as possible. For instance, standards and processes relate to system functional safety; intermediate data can be grouped as parameterization. Only if the system functional safety has been verified, the other relative performance evaluations make sense.

6.2.1 Validation of the CPN model

In safety critical systems, design validation is an essential step that is required before any further analysis. Validation is a process of determining that a system actually can fulfill the purpose for which it was intended. In other words, validation is solving the issue of "are we building the right thing?" [78].

Here four system safety criteria of the CPN model are validated. The validation process provides evidence whether the model satisfies specific requirements, which ensure the functional safety evaluation in Section 4.2. In this study, we apply the state space method to carry out the model validation [71]. The model has to satisfy the following aspects as shown in Table 6.2. V1 guarantees that all states are involved in the state space, V2 reflects the abnormal system termination; V3 is an indicator of the endless loop in the system; V4 contributes to validate the simulation accuracy.

Table 6.2: Validation requirements and methods

Requirements	Description
V1: All available states are calculated	To implement state space analysis, a full state space needs to be calculated, which means all system available states are involved.
V2: The model runs infinitely	The TTDMS should operate endlessly. Hence, unreasonable dead markings are not allowed.
V3: Self-loop is forbidden	Self-loop means a system holds on the same marking through an executable transition, and it consumes additional system resources.
V4: Simulation accuracy	Parameterization applied to figure out the system performance, the simulation results' accuracy should be validated.

In the descriptions, marking is a representation of tokens in places that constitute a specific combination [79]. Dead marking is a unique marking; it reflects that the system is terminated, and no further subsequent actions are available. Unreasonable dead markings are undesired. Self-loop can be a design defect, which should be avoided and corrected, or it can also be an essential update action in a system.

```
Statistics
------------------------------------------

 State Space
    Nodes:  16
    Arcs:   19
    Secs:   0
    Status: Full

 Scc Graph
    Nodes:  1
    Arcs:   0
    Secs:   0
```

Figure 6.3: State space analysis report of TTDMS CPN model

We apply these four validation requirements into the TTDMS. The result of V1 is presented in Fig. 6.3, which is the state space generation result of the CPN model. The status of the state space is Full, and it indicates that all available states have been calculated in the state space. Fig. 6.4 shows the state space, which has been omitted the labels on the arcs specifying the binding elements. There are 16 nodes and 19 arcs in the state space. It means that the whole system model can generate 16 different markings, and 19 arcs connect the markings. Scc (strongly connected component) Graph means every vertex is reachable from any other vertex in this graph, and it declares that every marking in the model is reachable from any other marking.

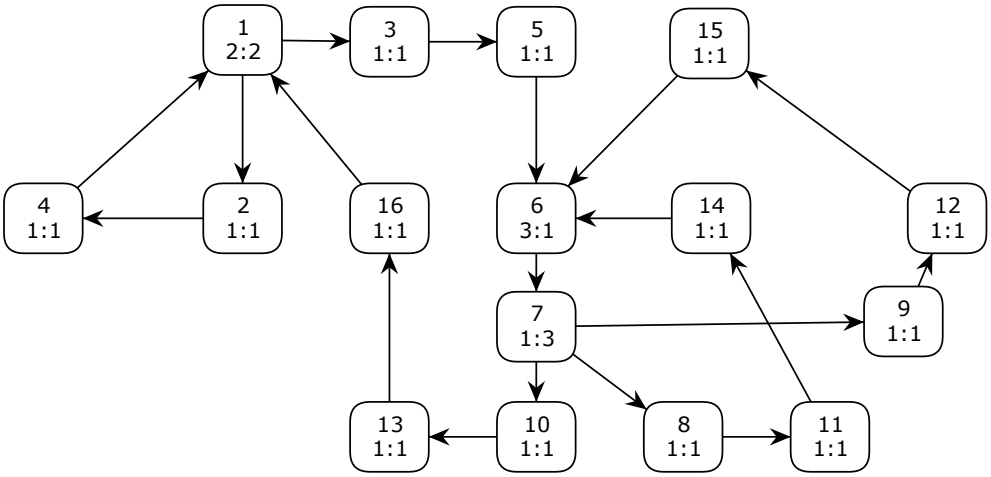


Figure 6.4: State space

The description of other validation requirements is drawn by Computational Tree Logic (ASK-CTL). The ASK-CTL is interpreted over the state spaces (also called the occurrence graph or reachability graph/tree) of the CPN model in the CPN Tools [80]. The ASK-CTL logic and model checker are implemented in SML (a general-purpose functional programming language), and queries are formulated directly in SML syntax. Checking queries written in SML are proposed to carry out various verifications. Some basic SML functions are introduced in section 5.2, and more details are available in the reference [72].

Checking of V2 and V3 can be achieved by the following queries as shown in Fig. 6.5. Function `SelfLoopTerminal` returns the nodes that has itself as the output. Function `ListDeadMarkings` calculates all the nodes that are dead and returns them as a list. In the result, it expresses that there are no self-loop terminals and no dead markings in the formal model. It indicates that the system will not stop at a permanent state, and there is no risk for resources consumption caused by the infinite invalid self-loop.

V4 validates the accuracy of mathematical simulation in CPN model. In order to validate the results, the time consumed by establishing a connection is assumed to follow an exponential distribution as shown in Fig. 6.6. The system returns to `1‘IDLE` after the connection is established, this choice aims at simplifying the analysis. As a result, it is obtaining quickly enough data to do statistical analysis. The result is shown in Fig. 6.6, the Probability Density Function of obtained online by simulation fluctuates around the theoretical result.

Checking queries	Result
fun SelfLoopTerminal n=(OutNodes(n)=[n]) fun InValidTerminal()=PredNodes(EntireGraph, fn n => (SelfLoopTerminal n), NoLimit); let val fid = TextIO.openOut "Logic verification results.txt" val _ = if InValidTerminal()=[] then TextIO.output(fid, "There is no self loop terminal!\n") else TextIO.output(fid, "List of self loop terminals: \n") val _ = EvalNodes(InValidTerminal(), fn n => INT.output(fid,n)) val _ = TextIO.output(fid, "List of dead markings: \n") val _ = EvalNodes(ListDeadMarkings(), fn n => INT.output(fid,n)) val _ = TextIO.output(fid, "\nNumber of dead markings: ") val _ = INT.output(fid,length (ListDeadMarkings())) in TextIO.closeOut(fid) end	There is no self loop terminal! List of dead markings: Number of dead markings: 0

Figure 6.5: Self-loop terminal and dead markings in the CPN model

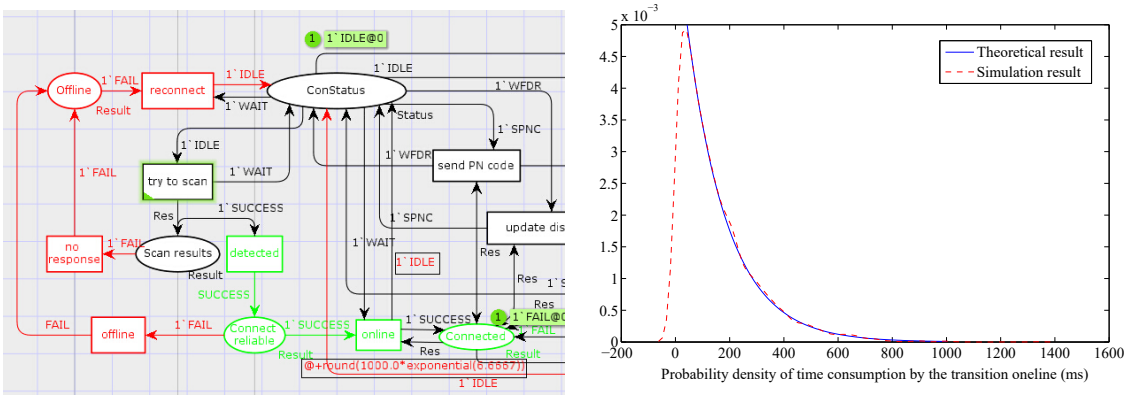


Figure 6.6: Simulation accuracy: parameter setting in CPN model and simulation result

6.2.2 Functional safety verification of TTDMS

In this section, the verification process adopts a universal approach. Hence, this process can also be used to analyze different sorts of systems. The procedure is carried out by a state space analysis to check the requirements. This procedure is about checking whether a particular state or variable meets given specifications, which reflects the feasibility of the system and predicts whether the system can run normally. In this case, parameterization is not considered so as to limit the number of system states. Based on the state space analysis, the system's structure correctness is validated.

From the definition of functional safety given by the International Electrotechnical Commission (IEC), functional safety means the system will operate correctly with response to

its inputs [71]. Potential hazardous conditions can be precluded by applying the functional safety analysis. To carry out the functional safety verification, three representative requirements are presented here. These requirements can be used: (i) to determine system resource consumption; (ii) to traverse and identify required system states; (iii) to implement reachability analysis.

These requirements can be checked by analyzing the reachability graph. This graph contains all the possible occurrence sequences and reachable markings. The relationship between system functions and reachability graph is shown in Fig. 6.7. Importantly, the following elements details are involved.

- Path. A system transfers from the idle state to a required state, it can be reflected by existing a path transferring from the source marking to the target marking.
- Available markings. Different states can directly activate a system target marking. The states are expressed by available markings, such as A1, A2, and A3.
- Token. The token details of a marking can also be analyzed. For instance, the token details of target marking are shown in the rectangle in Fig. 6.7.

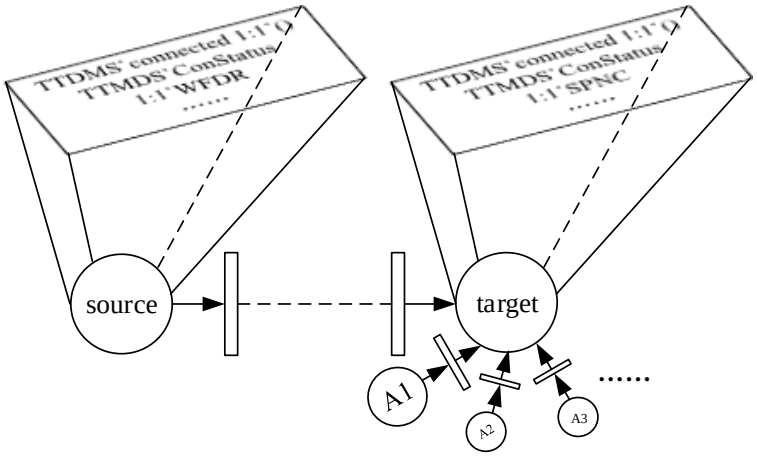


Figure 6.7: Mapping between system function and reachability graph

Hence, in a CPN model, the three safety properties requirements mentioned before can be described respectively as:

- State verification:
  - ensuring all reachable states satisfy a given prediction on states;
  - determining whether a reachable marking exists that can fulfill a given prediction;
- Function verification:

– ascertaining whether a reachable sequence that satisfies a particular state prediction exists;

– ensuring one state can and can only be modified by one event.

For the state verification, the following three examples are illustrated, the meaning of places and tokens in the CPN model are shown in Table 6.1.

(i) The `ConStatus` indicates the system main state, and it controls the system actions. The place `ConStatus` can and only can hold one specific value at one time. It means that $\forall M \in \mathcal{R}(M_0) : |M(ConStatus)| = 1$ where $|M(ConStatus)|$ denotes the size of the multiset $M(P)$. The initial state is 1‘IDLE. It is obvious that there must always be one and only one token on the place ConStatus, because a system can only work in one particular situation at any given moment. Here boundedness properties can be applied to check this function.

Generally, the boundedness properties specify how many and which tokens a place holds [69], it can reflect hardware or software resource requirements, since one place holding more tokens indicates more storage space is required. The checking query in Fig. 6.8 is available to verify the boundedness properties. Function `UpperInteger` calculates the maximal number of tokens on the place `ConStatus`. The result is 1, which means that at most one state is available at the given time. Function `LowerInteger` is similar to function `UpperInteger` but reflects the minimum number of tokens on the place. The result is 1, which indicates that there is always one state of the system. Function `UpperMultiSet` indicates the details of all available states for the TTDMS. It is clear that TTDMS has four operation modes as expected.

Checking queries	Result
let val fid = TextIO.openOut "Boundedness.txt" val _ = TextIO.output(fid, "system main statue: \n") val _ = TextIO.output(fid, "Upper integer: \n") val _ = INT.output(fid,UpperInteger (Mark.TTDMS'ConStatus 1)) val _ = TextIO.output(fid, "Lower integer: \n") val _ = INT.output(fid,LowerInteger (Mark.TTDMS'ConStatus 1)) in TextIO.closeOut(fid) end	System main statue: Upper integer: 1 Lower integer: 1

Figure 6.8: Boundedness properties of place

As shown in the results, both the upper integer and lower integer are 1 on the place `Mark.TTMS‘ConStatus`. It suggests that the model operation obeys the expected behavior.

(ii) Additionally, the main system states must be limited to the proper state in specific situations. For instance, before the connection link is established, the main state should be only in `IDLE` or `WAIT` [3]. It illustrates that the system will hold a proper state in a specific situation. Variable `Offline` involves all the nodes when TTDMS is offline. In this kind of situation, TTDMS should be in a proper main state. As shown in the Fig. 6.9, the tokens on the place `ConStatus` are `1‘IDLE` and `1‘WAIT`.

Checking queries	Result

val Offline=PredAllNodes(fn n=>Mark.train_control'connected 1 n=[]);
let
val fid = TextIO.openOut "Proper state.txt"
val _ = TextIO.output(fid, "When the connection is offline, the main state is: \n")
val _ =TextIO.output(fid, STRING.mkstr_ms(EvalNodes(Offline, fn
n=>st_Mark.train_control'ConStatus 1 n)));
in
TextIO.closeOut(fid)
end
	When the connection is offline, the main state is:
	1`"train_control'ConStatus 1: 1`IDLE"++
	3`"train_control'ConStatus 1: 1`WAIT"

Figure 6.9: Token detail on a place

(iii) After the connection between two TTMDS is established, the distance information update starts after the following train's TTMDS sends PRN code. It is reflected in the CPN model as that markings obtaining the token `1‘SPNC` should be found, and parallel arcs which can transfer from other markings to it should also be enumerated. This feature can be performed by the checking query in Fig. 6.10. Function `Operating(SPNC)` returns all the nodes with token `1‘SPNC`. The results show that the marking 6 contained the required token, and three arcs are connecting to the marking 6. The details of the state space graph are shown in Fig. 6.11. It is clear that after transitions `TTDMS’WFND`, `TTDMS’update_distance`, and `TTDMS’ONLINE`, the system turns to send PRN code procedure. These three transitions correspond with three actions in the TTDMS: wait for the data receiving, update distance, and establish the connection, respectively.

In general, this query can be used to check the following function: when one system can turn into a particular state, and which transitions can trigger it. It is a fundamental feature of the functional safety verification. The reachability of $M_m[\sum Y > M_n$ can be described as tokens' consumption and generation.

To illustrate the function verification, the following two cases are given:

(i) In the calculation process, the main steps are implemented through transitions `send PRN code` and `update distance`. Transition `send PRN code` can only be activated when the connection is established. When the `R_Message 1` obtains full package (`1‘FULL`), the

[3]See Table 6.1

Checking queries	Result

fun Operating(p:Status)=PredAllNodes (fn n=>cf(p,Mark.TTDMS'ConStatus 1 n)>0);
Operating(SPNC);
val L=EvalNodes(Operating(SPNC), fn n=>InArcs(n));
val fid = TextIO.openOut "RelativeBoundedness.txt"
val _ = TextIO.output(fid, "Markings contain token 1`SPNC are: \n")
val _ = EvalNodes(Operating(SPNC), fn n => INT.output(fid,n))
val _ = TextIO.output(fid, "\nArc Details: \n")
fun listpro(LL)=
let
val OR=LL;
val LL=List.nth(LL,0);
val test=InArcs(LL);
val _ = EvalArcs(test, fn a => INT.output(fid,a));
val _ = EvalArcs(test,
fn a => TextIO.output(fid, st_BE(ArcToBE(a)) ^"\n"))
val RR=List.drop(OR,1);
val _=if RR<>[] then listpro(RR) else TextIO.output(fid, "Here is end!\n")
in
TextIO.closeOut(fid)
end;
listpro(Opreating(SPNC));
	Markings contain token 1`SPNC are:
	6
	Arc Details:
	18 17 6 TTDMS'WFND 1:
	{Res=SUCCESS}
	TTDMS'update_distance 1:
	{Res=SUCCESS}
	TTDMS'online 1: {Res=FAIL}
	Here is end!

Figure 6.10: Query to analyze places and arcs related to a particular token

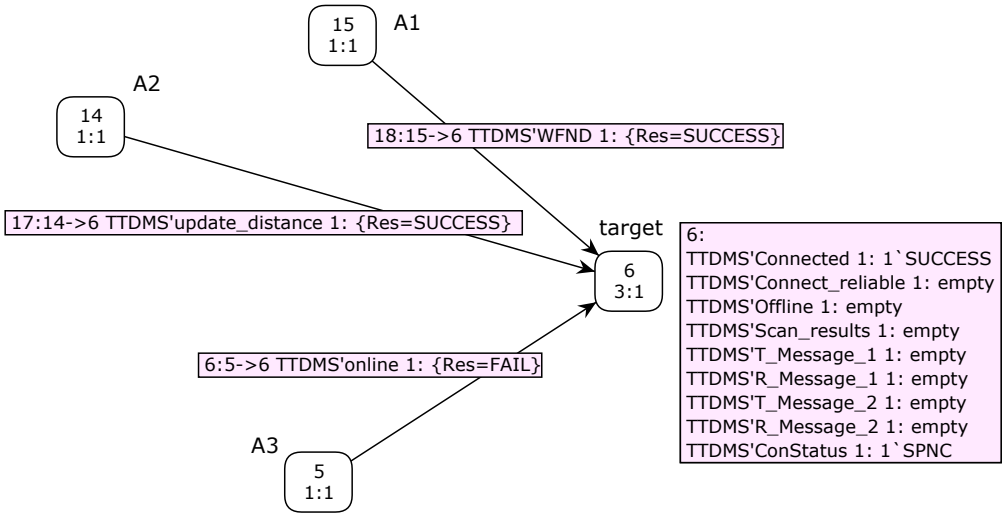


Figure 6.11: Details of the checking result in state space graph

transition `update distance` is triggered. The checking query in Fig. 6.12 can be used to implement this validation requirement. Function `TI_Arc` observers all the arcs that have connections to the transition `update distance`. Variable `ArcList` represents the results. It is clear to notice that the system will send PRN code when the connection is successful (`1‘SUCCESS`). All the markings that can trigger the distance update are with a `1‘Full` package received.

Checking queries	Result

let
val fid = TextIO.openOut "Event_test.txt"
val _ = TextIO.output(fid, "Arcs relate to transition:\n")
fun TI_Arc(TI:TI.TransInst)=PredAllArcs(fn a=> ArcToTI a=TI);
val ArcList=TI_Arc(TI.train_control'send_PN_code 1);
val _ =EvalArcs(ArcList,fn a=> INT.output(fid,a));
val _ =TextIO.output(fid, "\nOriginal Nodes:\n");
val NodeList= EvalArcs(ArcList, fn a=>SourceNode(a));
val _ =EvalNodes(NodeList,fn n=> INT.output(fid,n));
val _ =TextIO.output(fid, STRING.mkstr_ms(EvalNodes(NodeList,fn n=>
st_Mark.train_control'connected 1 n)));
val ArcList=TI_Arc(TI.train_control'update_distance 1);
val NodeList= EvalArcs(ArcList, fn a=>SourceNode(a));
val Enodes=PredNodes(NodeList, fn n=>Mark.train_control'R_Message_1 1 n=[FULL], NoLimit);
val _=if NodeList=Enodes
then TextIO.output(fid,"The nodes trigger update_distance are nodes with FULL package\n")
else TextIO.output(fid,"validation failed")
in
TextIO.closeOut(fid)
end
	Arcs relate to transition:6
	Original Nodes:5 1`"train_control'connected 1: 1`SUCCESS"
	The nodes trigger update_distance are nodes with FULL package

Figure 6.12: Function validation examples

(ii) When a disconnection happens, the system tries to establish the connection again and continues to detect the distance between two trains. The following method in Fig. 6.13 is applied to check whether the execution is available: after `R_Message 1` obtains `1‘NULL` (disconnection), the `ConStatus` can change to `1‘SPNC` again.

Checking queries	Result

fun Dataloss(p:PACK)=PredAllNodes (fn n=>cf(p,Mark.TTDMS'R_Message_1 1 n)>0);
fun Opreating(p:Status)=PredAllNodes (fn n=>cf(p,Mark.TTDMS'ConStatus 1 n)>0);
val Source=List.nth(Dataloss(NULL),0);
val Target=List.nth(Opreating(SPNC),0);
Reachable'(Source,Target);
if (Reachable(Source,Target))
then
print"reconnection established"
else
print"can not establish connection"
	val Dataloss=fn:PACK->Node list
	val Opreating=fn:Status->Node list
	val Source =16:Node
	val Target=6:Node
	A path from node 16 to node 6 is: [16,1,3,5,6]
	val it=true:bool
	Reconnection established val it=(): unit

Figure 6.13: Reachability analysis: system returns to a particular state from another

In general, this query can be assigned to check whether a system can transfer from a specific state to another. The query can do reachability checking more easily by directly configuring valuables `Source` and `Target` with relevant tokens

The result represents a possible path, it is `[16,1,3,5,6]`, as shown in Fig. 6.14. When the receiver receives no more data (`R_Message 1:1‘NULL`), the system tries to establish the connection again through transition `reconnect`, if the communication link is available (`Connected 1:1‘SUCCESS`), the system turns to the detection mode again (`ConStatus 1:1‘SPNC`).

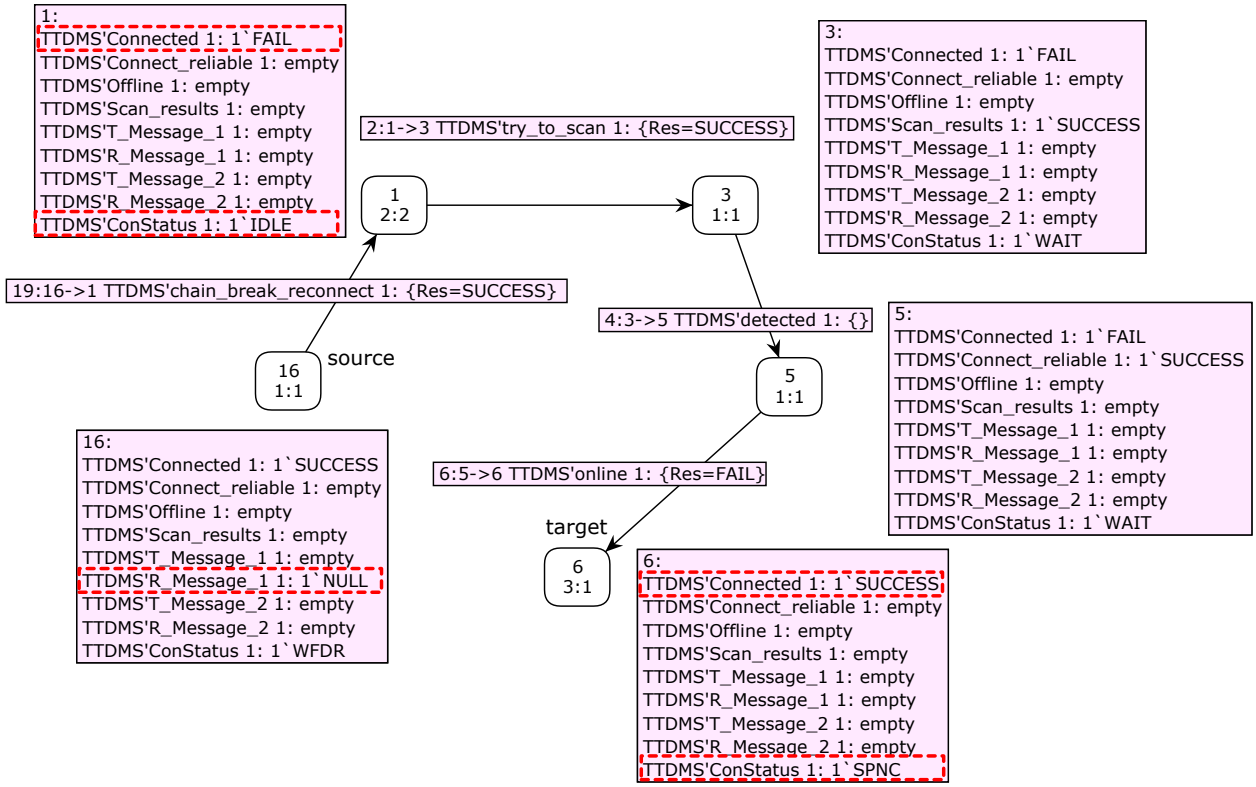


Figure 6.14: Details of the reachability analysis result

6.3 Performance evaluation of TTDMS

To analyze the system performance, we applied parameterization into the formal model. Each token can be parameterized with a required meaning in the CPN model. By adapting values to corresponding quantities, a system can be more easily understood and changed than it in executable codes. With the data obtained from the model, the system performance measurement is available. The data can be used to do statistical analysis, and responding safety and availability analysis can be carried out.

The TTDMS is based on the time delay to calculate the distance between two systems. Hence, the time delay's deviation has primary influences on measurement results. In this dissertation, the model is based on the following presuppositions:

- Hardware failure rates are not involved in the model, and the hardware reliability is estimated in section 4.3.
- Time delay caused by mathematical calculation error is not involved.
- The time consumption distributions are chosen based on our prototype machine experimental data and previous researches of Prof. Dr. Armin Zimmermann [76].

With parameterization applied in the CPN model, various stochastic distributions can be quickly configured and analyzed. Additionally, the model is readable and clear.

6.3.1 Parameters initialization

In section 6.2, model validation and system functional safety verification have been implemented, respectively. Only under this precondition, the parameterization can be conducted reasonably in the model.

Normally, the distance information update cycle is in sub-second level (in our prototype machine is 100 milliseconds). As proposed in section 6.1.2, there is a time delay existing between a new distance update and the last one [76]. For the railway transportation safety, investigations of the maximum time delay t_{max} and the possibility of applying the emergency braking or the normal braking caused are essential. To analyze the upper limit of t_{max}, the physical distance s is mapped to a time consumption level, as shown in Fig. 6.15 for an illustration.

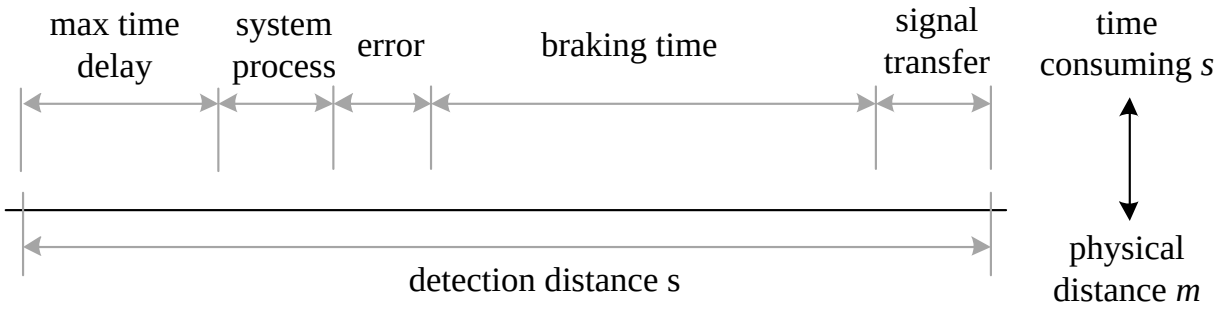


Figure 6.15: Time consumption and physical distance

Firstly, the time used for the signal transition in the air, which equals $\frac{2 \times s}{C}$, needs to be subtracted from the whole time consumption. Secondly, when the TTDMS calculates the results of the signal transfer time, the information is already late, because the hardware system needs time to finish mathematical calculations; and then there is a calculation error possible in the signal transfer time results, which relates to the device's ability. These time delays are denoted with Δt. Thirdly, the time used to finish the emergency braking needs to be subtracted as well. The braking distance L_{EB} depends on the actual train velocity v.

Hence,

$$t_{\max} = \frac{s - L_{EB}}{v} - \Delta t - \frac{2 \times s}{c} \quad sec \tag{6.1}$$

Here we choose the deceleration a is 1 m/s^2, which is the average value in practice [81], hence

$$L_{EB} \approx \frac{v^2}{2a} \approx \frac{v^2}{2} \quad m \tag{6.2}$$

$$t_{\max} = \frac{s - L_{EB}}{v} - \Delta t - \frac{2 \times s}{c} < \frac{s - L_{EB}}{v} \approx \frac{s}{v} - \frac{v}{2} \quad sec \tag{6.3}$$

The worst-case delay can be estimated as follows: after the disconnection, followed by the reconnection and transmission error. The first step is to establish the connection again. The system tries to build the communication channel when the place `ConStatus` holds the token `1‘IDLE`. Here it is assumed that if the connection was not available in 3 seconds under the surveillance mode, the connection establishment action retried. Based on the experimental results of the prototype machine, the time used to establish the connection is no more than 150 milliseconds. Hence, the corresponding distribution is thus exponential with a parameter λ. Here,

$$E(X) = \frac{1}{\lambda} = 0.15 \quad sec \tag{6.4}$$

$$\lambda \approx 6.6667 \quad sec^{-1} \tag{6.5}$$

Because of the transmission error results in the time delay, during this period, there is no useable data to update the distance until a full complete packet is received. Here the delay has been assumed to be memoryless and be smaller than x seconds in p of all cases. Here we choose $x = 5$ seconds, $p = 75\%$. Thus, the transition follows an exponential distribution with the expectation related parameter λ, where

$$p = 1 - e^{-\lambda x} \tag{6.6}$$

$$\lambda = -\frac{\ln(1-p)}{x} \approx 0.2773 \quad sec^{-1} \tag{6.7}$$

When it comes to the disconnection, this kind of state is not common in nowadays wireless communication networks. For instance, a disconnection occurs only 10^{-4} times per hour in the Global System for Mobile Communications (GSM) [76]. This parameter is different from specific communication technologies applied to the real system. Therefore, it is assumed that in 80% of all cases, the system attempts to establish the connection within not more than 1 second. Hence,

$$\lambda = -\frac{\ln(1-p)}{x} \approx 1.6094 \quad sec^{-1} \tag{6.8}$$

It is important to note that, the parameters vary in different communication technologies. In this section, they are only used for illustrating the performance of our approach.

6.3.2 Simulation and results

For an exponential distribution, the Probability Density Function (PDF) and Cumulative Distribution Function (CDF) express as $f(x) = \lambda e^{-\lambda x}$ and $F(x) = 1 - e^{-\lambda x}$, respectively. Hence, let T_1, T_1, T_3 are the three independent random variables assigned with exponential distributions with parameters $\lambda_i \, (i = 1, 2, 3)$. T_1, T_1, T_3 represent time consumed by disconnection, connection establishment, and transmission error, respectively. Hence, the whole time consumption is

$$t_{\max} := T_1 + T_2 + T_3 \quad sec \tag{6.9}$$

According to the mathematical character of exponential distributions in reference [82], we got

$$ft_{\max}(x) = \sum_{i=1}^{n} \frac{\lambda_1 ... \lambda_n}{\prod\limits_{\substack{j=1 \\ j \neq i}}^{n} (\lambda_j - \lambda_i)} e^{(-x\lambda_i)} = [\prod_{i=1}^{n} \lambda_i] \sum_{i=1}^{n} \frac{e^{(-x\lambda_i)}}{\prod\limits_{\substack{j=1 \\ j \neq i}}^{n} (\lambda_j - \lambda_i)} \tag{6.10}$$

where $x > 0$. Hence

$$ft_{\max}(x) = [\lambda_1 \cdot \lambda_2 \cdot \lambda_3] \cdot \left[\frac{e^{-x\lambda_1}}{(\lambda_2 - \lambda_1)(\lambda_3 - \lambda_1)} + \frac{e^{-x\lambda_2}}{(\lambda_1 - \lambda_2)(\lambda_3 - \lambda_2)} + \frac{e^{-x\lambda_3}}{(\lambda_1 - \lambda_3)(\lambda_2 - \lambda_3)} \right] \tag{6.11}$$

$$E_{T_1+T_2+T_3}(X) = \frac{\lambda_2\lambda_3 + \lambda_1\lambda_3 + \lambda_1\lambda_2}{\lambda_1\lambda_2\lambda_3} \quad sec \tag{6.12}$$

It is obvious that, compared with mathematical calculations, the simulation procedure in the time CPN model are much closer to the actual scenario, and it can solve more complex distributions without having an onerous computation. With the parameters (refer to the dotted rectangles) applied to the CPN model, timed based analysis is available, as shown in Fig. 6.16. For instance, in one dotted box (only used as a mark) `@+(1000.0*round(exponential(6.6667)))` means a random time delay that follows an exponential distribution with a parameter $\lambda = 6.6667$ is added after the transition has been fired.

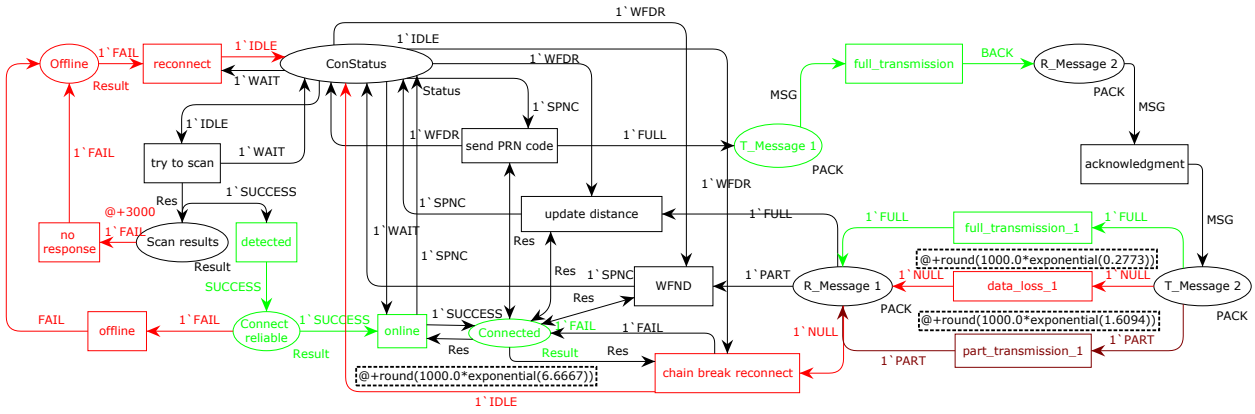


Figure 6.16: With parameterization in TTDMS CPN model

For 1,000,000 times Monte-Carlo-Algorithms testing (Fig. 6.17), the maximum result is 46.92 seconds in this one million times simulation. Based on the data of emergency braking curves, taking the CRH2-300 high-speed train in China as a case study, the emergency braking distance is 2786.68 meters when the train speed is 300 km/h [2]. To make sure all the 1,000,000 times results do not trigger the emergency braking, the TTDMS should have a detection distance s further than 6,695 meters. If our technology can only maintain the detection distance up to 5,000 meters, then the time delay must be smaller than 27

seconds.

$$t_{\max} < \frac{s - L_{EB}}{v} = \frac{5000 - 2786.68}{83.3} \quad sec \tag{6.13}$$

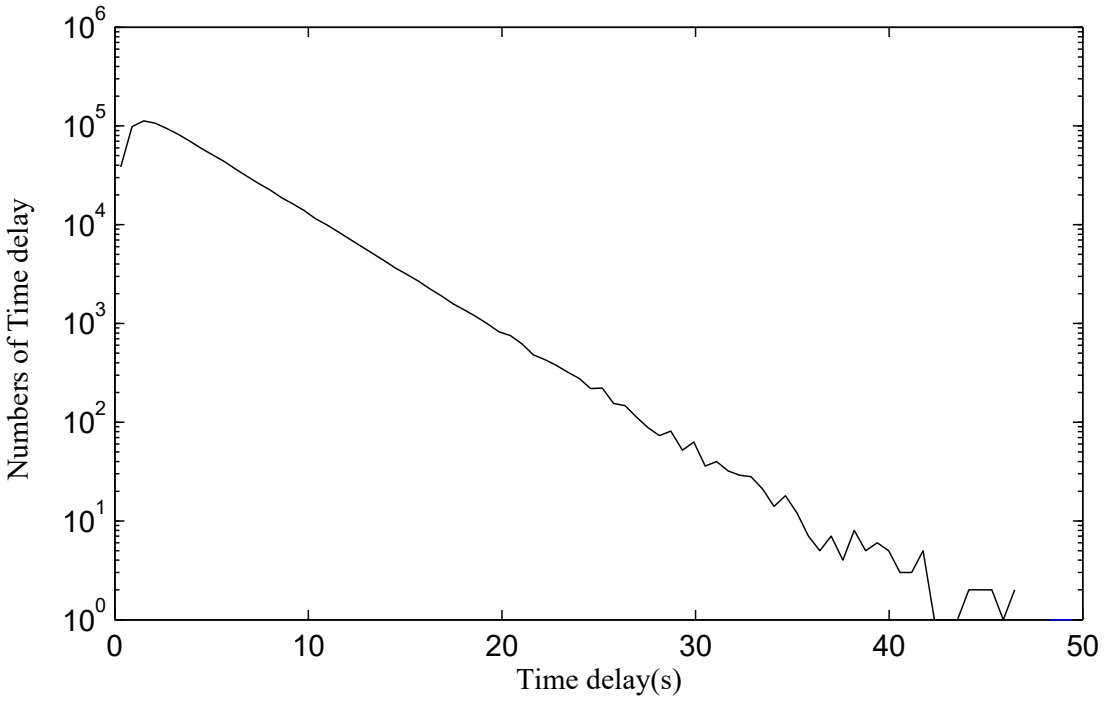


Figure 6.17: Distribution of time delay

Fig. 6.18 presents the intercorrelations among the TTDMS availability, detection distance, and braking strategies. If we continued to use this system factors with a detection distance 5,000 meters, there is a less than 7×10^{-4} probability to apply an emergency braking. However, the emergency braking is not a frequently used braking strategy. For a normal braking of CRH2-300 as an example, the braking distance is 5,172.28 meters. Hence, the TTDMS should have a distance detective ability farther than it in the emergency braking mode, or else the train speed has to be reduced if this system is applied in practice. If the system can detect 6,000 meters, the deadline is 9.94 seconds, and it has a 8×10^{-2} probability of actualizing normal braking.

$$t_{\max} = \frac{s - 5172}{83.3} - \Delta t - \frac{2 \times s}{c} \quad sec \tag{6.14}$$

All in all, based on simulation results we can: (i) evaluate the actual TTDMS performance,

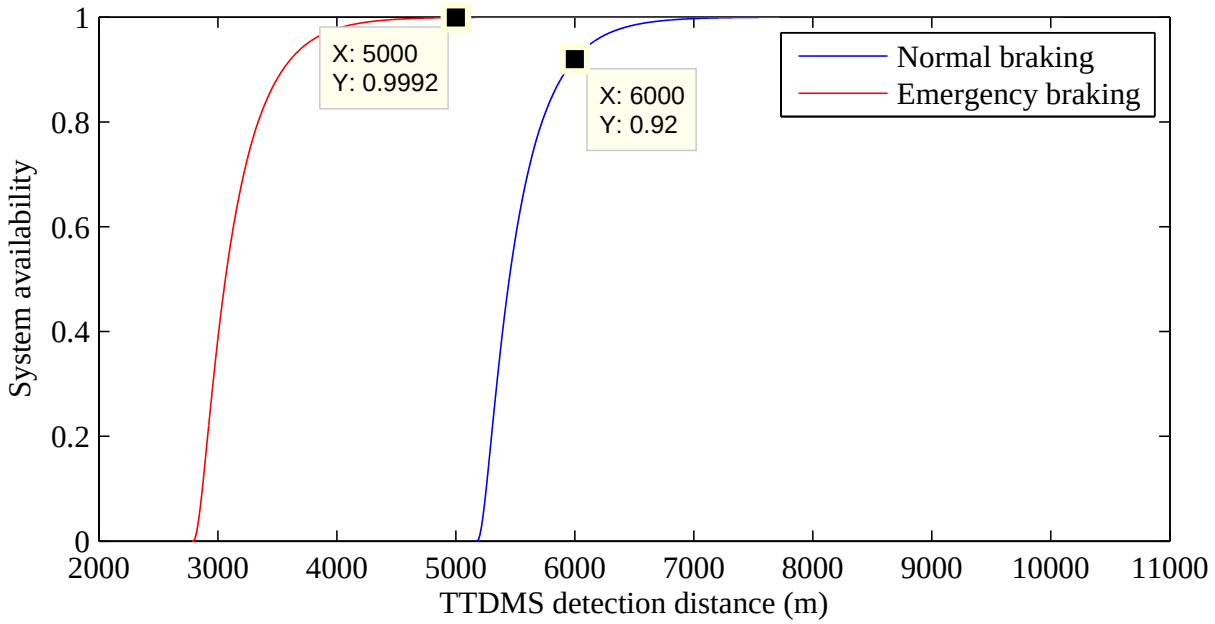


Figure 6.18: Intercorrelations among TTDMS availability, detection distance, and braking strategies

such as the probability of triggering an emergency braking and a normal braking; (ii) adjust the TTDMS performance to meet engineering requirements. For instance, braking distance, train operating velocity, and measurement accuracy.

6.4 Summary

In this chapter, it has been shown how CPNs is applied to evaluate and verify a system. Different from most analysis methodologies, a novel approach has been introduced to carry out both functional safety evaluation and simulation-based verification. This procedure was illustrated based on a TTDMS. Based on the system property concept, the TTDMS was formalized in the form of CPN model. Using state space analysis and simulation-based verification, it proved that there are no dead marking and self-loop, and the mathematical simulation is accurate. Verifications in both system functional safety and performance were implemented. Different checking queries were presented to analyze the system functional safety. The results indicated that the TTDMS could operate correctly with response to its inputs and conditions. Based on this precondition, three parameters (fail to establish a connection, transmission error, and disconnection), which affected the system performance, had been discussed and parameterized in the CPN model. The simulation results indicated

the intercorrelations among TTDMS availability, detection distance, and braking strategies. It was found by the functional safety and simulation-based analysis that the whole procedure was highly efficient and was easier to implement than mathematical calculations. The further work is that one can propose a better organized and thorough checking process in the formal model analysis, and introduce different checking queries more systematically.

7 Model based code generation and time estimation

In this chapter, a safe software code framework is generated from the model structure, and combined with the hardware structure to assemble the actual system. Model-based software development is a primary approach to describe software on different levels of abstraction. Even though formal methods are not widely used in the development of software, formal models can prove the completeness and correctness of system requirements and specifications. Hence, inconsistencies and bugs can be avoided and predicted at an early stage of the development process.

7.1 Code framework generation from formal model

When the computerized executed model is composed and applied to evaluate the system's performances. It can be used to a code framework, and to do the model based estimation. The potential methods for the code generation from CPNs are structural analysis, simulation-based, and *OG* based. The structural analysis is to search for regular structures, such as loops and if-constructs, in the underlying graph. However, directly translatable graph patterns cannot be found in every model. The simulation-based approach is not suffering from the above problem, but its performance limits the application. What is more, the large memory footprint make this approach not suitable for embedded systems [83]. The *OG* based approach proposed in this paper does not have the efficiency problems, as the *OG* is the representation of all possible states of the model. Additionally, this approach is independent of the programming code. Hence, the *OG* is used to control the execution of the program and determine the next system state. An extensive review in the area of automatic code generation from Petri nets is given in [84].

A primary requirement of software design is the simplification, which makes the program convenient to read and maintain. The logic connection among different functions must be well

designed. It is conducive to ensure that the program is well organized and easily maintained. Improperly `jump` and `call` functions can cause system errors. In order to eliminate system bugs, the program structure should be outlined and proofed before the coding process.

In this section, we propose a procedure to organize various functions by using CPNs. As different systems require different programming languages, there will be no further elaboration on the specificities other than a brief mention of the encoding framework. The occurrence graph is able to generate codes efficiently [84]. The graph includes all available system states of the CPN model. The states are connected by corresponding transitions. All transitions in the graph represent different system events. The events can be handled as various functions in programming codes. Each arc represents a transition in the graph. In combination with the CPN model, the functional requirements are clear. Note that, not all the places are parameterized in this procedure. Parameterization can cause the state explosion of the occurrence graph when a value is stochastic. Here, we only take care of the system structure and no parameterization is required. The parameterization simulation procedure was carried out in section 6.3.

In software designs, state change relates to variables update. In Fig. 7.1, the same state (local state a) can be reached by different events (event P and event N). Logical states are not states of physical components, but "memory" states of abstract elements, which must be derived from the contextual domain knowledge and the objectives of the process [85].

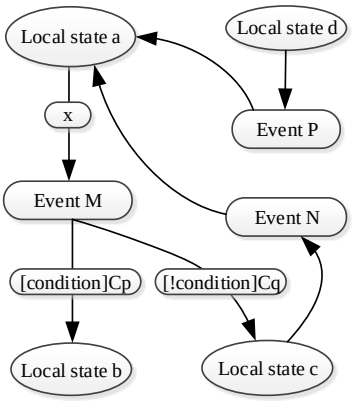


Figure 7.1: Function formulation

As shown in Fig. 7.2, the CPN model has the following four main factors: place, colored sets, transition, and node. Colored sets indicate different types of `Class` in program codes. A `Class` is a particular construct, and it groups variables and methods defined by users. Places represent local states that can be defined by variables. Transitions in the CPN model are functions or events in code. Transitions will cause a change in the state of a system,

which reflects as node changing in the CPN model. CPNs provide different color sets, which can be mapped onto a `Class` in the encoding process. The system development is based on models as an abstraction of the real world. An extensive review of existing work in this area of automatic code generation from Petri nets is given in [84].

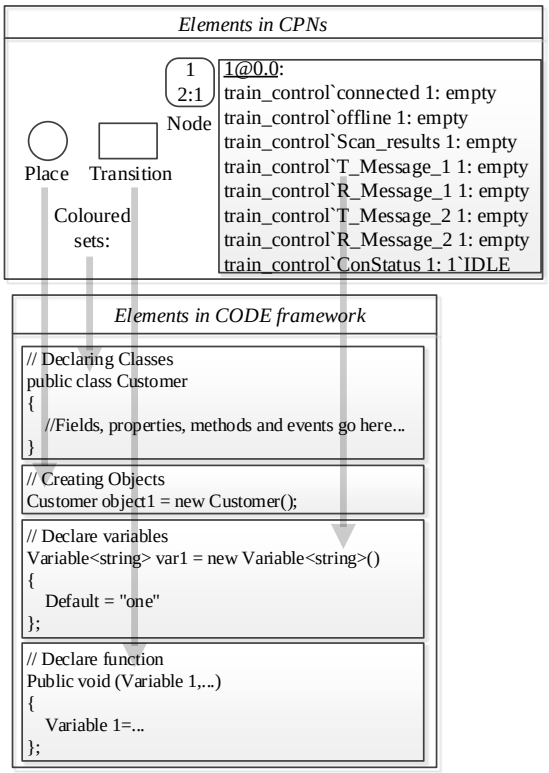


Figure 7.2: Elements in CPNs and code framework

The system operation is reflected as changing variables under specific events ($e_i \in E$). It can be given by $V \times E|G = V'$, where V, E, and G are the sets of all valuations of variables, events, and guard functions, respectively; V' is the new set of V after the changing. The values of variables are in valuation V, if event E occurs and guard G is true, then the new values for variables are given by valuation V'. Each event will have a guard function $g \in G$, which will decide the function results. The guard functions can be implemented as `Switch` function in codes. One state (local state a) can have different outputs (local state b and local state c) through one event (event m) as shown in equation 7.1. Meanwhile, different states can have the same output through different events as shown in equation 7.2.

$$(V, E, G) = V_a \times e_m|g = \begin{cases} V_b & g = c_p \\ V_c & g = c_q \end{cases} \tag{7.1}$$

$$\left[\begin{array}{ccc} V_m & V_q & \cdots \end{array}\right] \times \left[\begin{array}{c} e_m|g \\ e_q|g \\ \cdots \end{array}\right] = V_a \tag{7.2}$$

where $V_a, V_b, V_c, V_m, V_q, \cdots \in V$, $e_m, e_q, \cdots \in E$.

Fig. 7.3 indicates the full occurrence graph of the CPN model. Taking a partial graph for example, M_1 can transfer into M_2 and M_3 through the same transition `try to scan`. M_2 and M_3 indicate the success (`1‘SUCCESS`) and failure (`1‘FAIL`) of the scanned result, respectively. Marking M_3, M_9, and M_{10} can change into M_5 through transitions `ONLINE`, `update distance` and `WFND`, respectively.

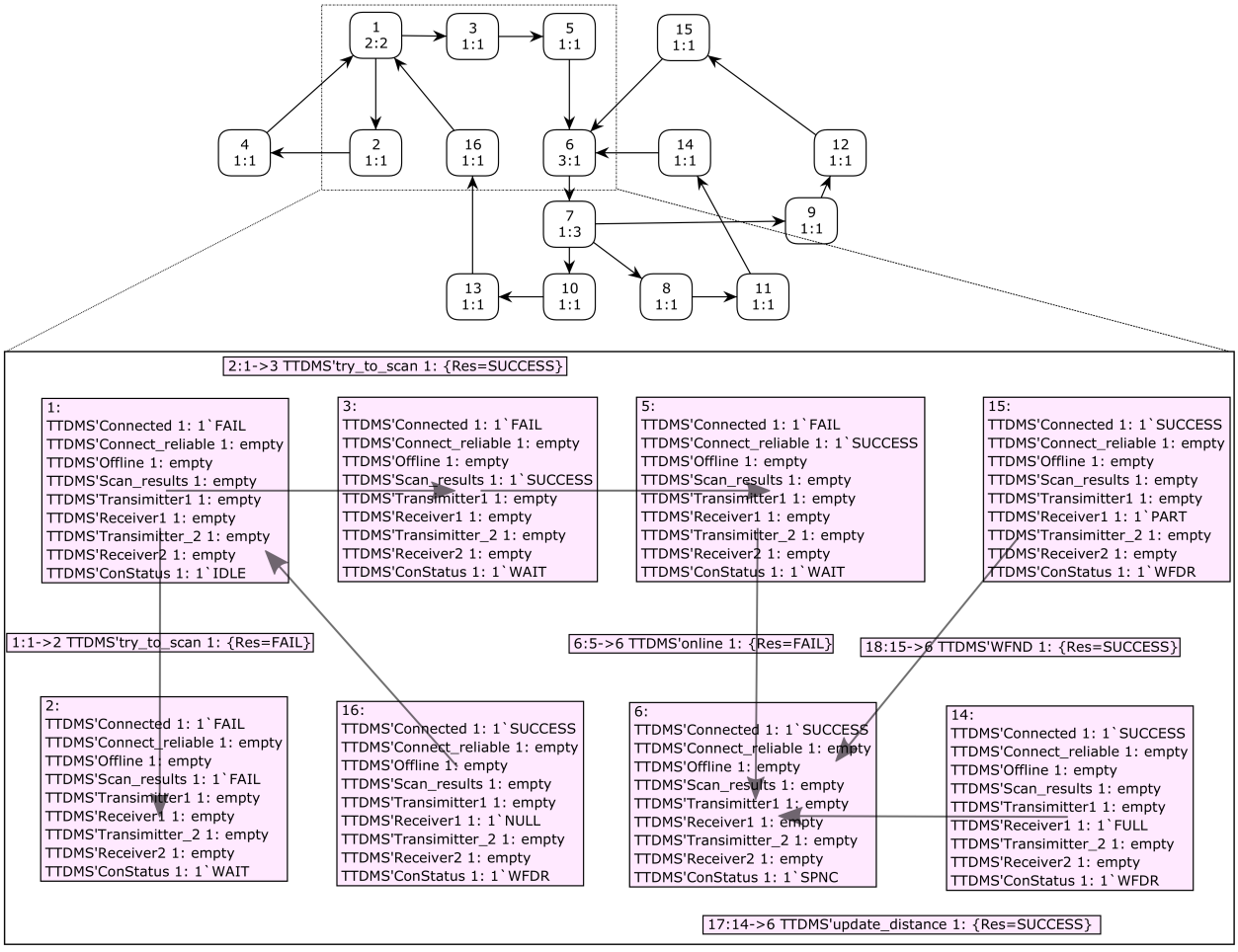


Figure 7.3: Occurrence graph and partial details

Hence, the procedure to generate a code framework can be indicated as shown in Fig. 7.4. After the OG is loaded, the token details (obj) in different V_m become clear. `foreach (colset`

`in` Σ) declares different `Classes` based on the **colset**. `foreach (place in` P) creates objects, which represent places in CPN model, based on respective `Classes`. `Initialization` (V_o) sets the system with initial variables. V_m can only be modified with a given g. As a result, L, a list of e_m, is available for generation, and it can be used to organize the encoding framework and do the execution time estimation.

Algorithm generate code framework

Load OG
$OG = \{V, E, G\}$
$L = \{\ \}$
$obj \in V_m \in V$
$V_o \in V$
foreach (**colset** in Σ)
  //Declaring Classes
  public class **colset**
  {
    //variable definition
  }
foreach (**place** in P)
  //Creating Objects
  colset obj =new **closet**()
V_m = Initialization (V_o)
foreach (OutArcs in V_m)
  $e_m = Arc \rightarrow TI.TransInst$
  $g = Arc \rightarrow Bind.Elem$
  if $V_m \rightarrow obj == g$ **then**
    $V_m = V_m \times e_m \mid g$
    add e_m **to** L
    return V_m, L
  endif

Figure 7.4: Model-based code-generating algorithm

With the same OG in one project, a standard way is available for different programmers to rely on to build and deploy applications. The OG helps programmers visualize the system structure. Unlike in standard applications, the system's control flow is not determined by the programmers but by the framework.

7.2 Model based execution time estimation

Although hardware accuracy, multi-path propagation and NLOS play indeed a major role in the measurement resolution, the software implementation could also have an impact, for example through the processing time induced by the source code configuration obtained with the CPN-based method.

In the analysis of many systems, it is necessary to know the execution time of the tasks in the system. Traditionally, the estimation has been calculated based on static analysis techniques. The source code or disassembled binary executable of the task is analyzed to determine the time required for the longest path through the code [86]. Given the code framework is generated based on the OG, which can be used to do the static analysis.

As shown in Fig. 7.3, after the `connected` is `SUCCESS`, it requires TTDMS five steps to finish one distance update. Equation (7.3) illustrates this:

$$V_t = V_s \times \prod_{m \in L} e_m | g \tag{7.3}$$

$$L = \left\{ \begin{array}{c} \texttt{send PRN code} \\ \texttt{full transmission} \\ \texttt{Acknowledgment} \\ \texttt{Link situation} \\ \texttt{update distance} \end{array} \right\} \tag{7.4}$$

where V_s and V_t represent variables in the source and target states, respectively.

The time required to execute an event is fit to the Gumbel distribution, $T_{e_m} \sim Gumbel\left(\mu_m, \sigma_m\right)$ [87, 88]. The parameters of a Gumbel can be estimated by applying the algorithm proposed in [86]. Here we give a proposal to carry out the estimation while no numerical parameters are considered.

In an action period, an event e_m is executed ρ_m times, for $\rho_m \in \mathbb{R}$. Hence, the total time taken is $W = \sum_{m \in L} \rho_m T_{e_m}$. The characteristic function $(\phi_W(t))$ and Probability Density Function $(f_W(t))$ of W are defined by equations (7.5) and (7.6) respectively.

$$\phi_W(t) = \prod_{m \in L} \left\{ \Gamma\left(1 - it\sigma_m\rho_m\right) e^{it\mu_m\alpha_m} \right\}, t \in \mathbb{R} \tag{7.5}$$

$$f_W(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \exp\left\{ it\left(\sum_{m \in L} \rho_m\mu_m - t \right) \right\} \prod_{m \in L} \Gamma\left(1 - i\rho_m\sigma_m t\right) dt \tag{7.6}$$

where $\phi_W(t)$ and $f_W(t)$ represent the results of characteristic function and PDF, respectively; Γ is the gamma function; $i^2 = -1$ is the complex unit. The details for the mathematical proof can be found in [88].

The CPN model reflects the system structure. As long as proper parameters are available, the system performance can be evaluated using a CPN model. However, it is not further discussed in this thesis.

7.3 Summary

In this chapter, a methodology for an encoding framework premised on an occurrence graph was introduced. This *OG* based approach permitted an efficient code framework generation. As a result, it offered an approach to convert the safe model to safe code framework. What is more, the encoding framework was applied to perform a static analysis of the execution time estimated in the latter part of this chapter.

8 Evaluation of the collision fault tree by means of CPNs

Probability-related system states can be analyzed with the Fault Tree Analysis (FTA). However, restricted by the commercial tools, the FT is limited to assess dynamic systems without event-repair operations and probability-related attributions.

Thus, this chapter proposes a new method to represent and extend the FT in CPNs, which combine event maintenance components, model correctness verification, time factors and mathematical calculation together. Additionally, it can be reused for customizations. The accuracy of the approach is verified by using model-based simulation and state space analysis. The performance and benefits of the new approach are demonstrated by investigating train to train collision failure models. The results indicate that involving model correctness checking and repairable events into modeling approaches can provide much more essential information than the traditional Fault Tree Analysis. With the collision failure model proposed in section 3.3, the dependability of the overall MA+ can be analyzed by means of this method.

8.1 Introduction

Despite developments in automation technology, system faults can exist at any time and in any situation. It is essential to evaluate the dependability of systems for the sake of maintaining an equivalent or a higher safety level, after a new system is involved.

Fault Tree Analysis (FTA) can qualitatively and quantitatively evaluate the reliability, and represent the relationship between different faults. However, in the general FTA, the constant failure rates, absence of repair events, and time duration limit the analysis ability of FT [62,89]. Some extended FTs are thereby proposed. For instance, publications [90,91] present a Multi-state Fault Tree, which involves repair events. Publication [92] introduces a Temporal Fault Tree, and it allows addressing dynamic behaviors that depend on time duration.

There is some commercial software, such as, Windchill FTA, ITEM ToolKit, Fault Tree++, and so on. They can provide various functions of event-oriented analysis. However, the limitations of commercial tools used for fault tree analysis restrict the application of the fault tree method. More importantly, for the dependability analysis of safety critical complex systems, some mathematical calculation, which can not be fully satisfied in the commercial software, is essential to quantify the dependability characteristic. Hence, it is necessary to provide an efficient method satisfying the following requirements (R):

- **R1**: take different failure rates into account;
- **R2**: demonstrate the time attributions;
- **R3**: carry out the mathematical calculation;
- **R4**: consider repairable multi-state components;
- **R5**: verify the model's correctness.

Hence, it is necessary to develop a methodology that can cover all these five requirements. These requirements are essential for improving the flexibility and continuity in system dependability analyses. Importantly, this solution is free and modifiable for special applications. The methodology is applied to describe practical systems, and do qualitative and quantitative analyses.

The remainder of the chapter is organized as follows: First the terminological relationships between FT and CPNs is discussed. The main contribution of this work is discussed in sections 8.3 and 8.4. Section 8.3 is dedicated to discussing the modeling process; the model's calculation accuracy and structure correctness are verified. For the sake of illustrating our process performance, based on the MA+ proposed in section 3.2, the train to train collision accident is selected as the top event in fault tree. Moreover, the evaluation approach is illustrated in section 8.4, which carries out the model correctness evaluation and system dependability analysis. Finally, section 8.5 presents the conclusion and future works.

8.2 Terminological relationships between FT and CPNs

Formalization & modeling can efficiently and cost-efficiently represent a real-world system. The system safety analysis based on modeling is widely used in different research areas. The description refinement of a system depends on the formalization degree. The higher the

formalization level used to describe the real system, the greater the possibility to mathematically verify the formalized concept system [39]. CPNs as high-level Petri nets are applied to evaluate the FT in this chapter rather than low-level Petri nets [93]. The components of (Colored) Petri nets and Fault Tree are shown in Fig. 8.1.

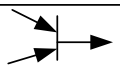
(Colored) Petri nets			Fault Tree		
Items	Symbol	Dynamic meaning	Items	Symbol	Dynamic meaning
Place		State	Fault Tree event		Fault
Transition		Event	--	--	Failure
Directional arc		Connect & control	No direction arc		Connect
Token	•	Actual state	--	--	--
Structure		Relation	Gate/logic		Relation

Figure 8.1: (Colored) Petri nets and Fault Tree

In FT, an "event" indicates an actual subject or case in a system, such as that the wayside equipment is out of service. However, "event" that in Petri nets has a dynamic meaning, and is represented by a transition. Transition is used to modify the system state, which is illustrated by the place in Petri nets. However, the failure action of the fault is not modeled and has not symbol in FT. Hence, in order to distinguish the terminological difference of "event", P_event and F_event are applied to represent the "event" meanings in Petri nets and Fault Tree, respectively.

What is more, the arcs in Petri nets not only have the connect but also control functions. However, the arcs in FT only play an indicative role. Importantly, CPNs provide different color sets, which are suitable to represent different attributions, such as F_event state, name, and maintenance information. All these aforementioned attributions can not be satisfied in FT. Additionally, the structure in Petri nets is more flexible to change to represent the gate logic in FT.

Hence, using the CPNs is a suitable and efficient way to represent and analyze the FT. It has advantages of the aforementioned FTA for dependability analysis. Moreover, model correctness verification is carried out by applying state space analysis; F_events and conditions are

represented by time durations but not considered as instantaneous; a subnet of a F_event is proposed to implement other procedures, such as the maintenance.

8.3 New procedure for representing an FT in a CPN model

For the dependability analysis of complex timed systems, the analysis method should make up the deficiencies in FTA. This chapter proposed a new process that can convert the typical FT to a CPN model. After the conversion, the F_event maintenance process, model correctness, time attribution, and parameterization are fully satisfied with the CPN model.

A CPN model consists of net and color structures. The net structure reflects relations among different F_events and states. The color structure can be used to represent the F_events' and states' attributions. In the CPN model, the model correctness, time attribution, parameterization, and F_event subnet are fully satisfied.

8.3.1 Requirements reflect in CPNs

The core components of FT are fault F_events and logical gate functions. By involving CPNs, the FT is converted into the CPN model, which contains the five requirements mentioned in chapter 8.1.

As shown in Fig. 8.2: different failure rates (**R1**) and time attributions (**R2**) are implemented in the F_event initialization by applying set color structures, which are detailed in section 8.3.2. The mathematical calculations (**R3**) are carried out by corresponding functions; the maintenance process of the F_events (**R4**) is involved by adding subnets, which are presented in sections 8.3.3 and 8.3.4.

The model correctness (**R5**) is performed during the verification process by using state space analysis. All these aspects guarantee the fault F_event maintenance, time consumption, system structure correctness, and appropriate parameterization. With the assistance of CPN model, the dependability analysis of complex timed systems can be extended.

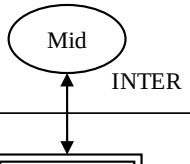
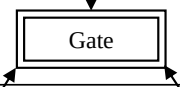
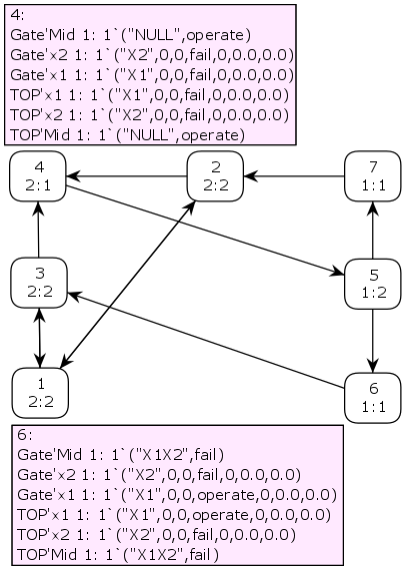
No.	Items	CPN net structure, state space	Description	Requirements
1	F_event		• F_event name • F_event state	--
2	Gate structure		• Fire/ solve the Mid event • Mathematical calculation	R3
3	F_event		• F_event name • Service time • Real operation time • F_event state • Periodic maintenance • Different failure rates …	R1,R2
4	P_event		• Periodic maintenance • Corrective maintenance	R4
5	State space		• Model correctness verification	R5

Figure 8.2: Requirements reflect in CPN net structure

8.3.2 Color set structure

A basic F_event must consist of these attributes: name, state, service time with relevant parameters, operation time, maintenance period, and lifetime [94]. An intermediate F_event is the output of a gate, and it has the name and state attributes.

A basic F_event has a name to identify itself. Additionally, the F_event can result either to faulty or to operate states. When the F_event is a fault caused by a failure, it will be repaired in a certain time. Since Weibull distributions are widely applied in practices, it

can describe many fault F_event's behaviors [95]. The Weibull distribution can be divided into a two parametric and a three parametric distribution. Except for parameters T and b, which represent the characteristic lifetime and the shape parameter, respectively, the three paramagnetic Weibull distribution exhibits an additional parameter t_0. It is the failure free time - location parameter. With this additional parameter, failures can be described that only begin to occur after a certain time t_0. While the two parametric Weibull distribution always describes failures starting from time $t = 0$.

In this chapter, we apply the two parametric Weibull distribution to predict the F_event's service time τ (statistical variable). As an intermediate F_event is the output of a gate, and it is solved or triggered based the gate logic and input F_events' states. Hence, the intermediate F_events have the name and state attributes.

```
colset subsystem=string;
colset timestamp=int;
colset servicetime=int;
colset state=with operate|fail;
colset maintenance period=real;
colset characteristic lifetime=real;
colset shape parameter=real;
colset FAULT=with
subsystem|timestamp|servicetime|state|maintenance
period|characteristic lifetime|shape parameter timed;
colset INTER=with subsystem|state timed;
```

Figure 8.3: Color set declaration

To represent the F_event's attributions mentioned above, the declarations of the color structure are presented as shown in Fig. 8.3. Here, color sets `FAULT` and `INTER` represent basic F_events and intermediate F_events, respectively. Note that, the color set can be modified based on an actual application. For example, the basic F_events with timestamps have the following attributions in `FAULT`:

- F_event name (string);
- real operation timestamp (int);
- service time τ (int);
- item situation (state);
- maintenance period T_{PM} (real);

- characteristic lifetime T (real);
- and shape parameter β (real).

8.3.3 Gate structures

AND and OR gates are the fundamental logic gates in FT. In many cases, only these two fundamental gates are needed to build a fault tree [41]. Here we only illustrate these two gates to introduce the gate structures in the CPN model.

An intermediate or top F_event happens when both F_events' state and time duration meet the gate logic. As shown in Fig. 8.4, F_event `X1` occurred at $n1$ period. The duration of the fault F_event relates to its maintenance time, and this fault F_event is activatable for the following steps during $[n1, n2]$. Hence, the time parameters are useful for the dependability analysis of technical systems [62]. As shown in the service fault F_events results, the AND gate has a lower probability to occur than the OR gate in the same period. Hence, to increase the system safety, the new system should combine with the existing F_events as AND gates.

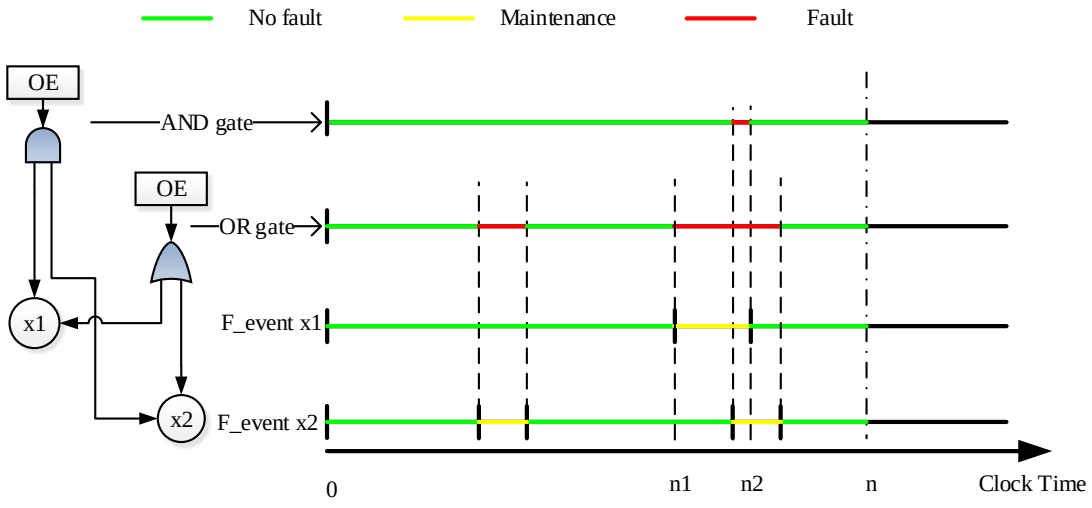


Figure 8.4: A example of occurrence of service failure

The basic net structure for the logical gate is introduced in Fig.8.5. In the repairable system, the fault will be corrected after it occurs. These two procedures are represented by transitions `solve` and `fire`, respectively. Hence, all the gates share the same net structure, which is flexible to be reused in the following model.

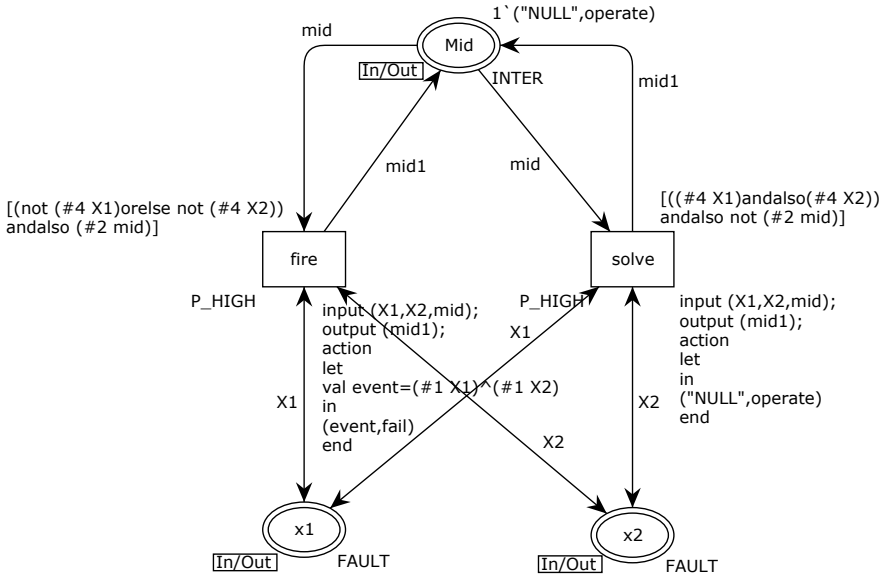


Figure 8.5: Gate structure in CPN model

Taking Fig.8.5, which indicates the AND gate, as a case study. When the F_events x1 and x2 failed in the same time period, the guard of `fire` is satisfied. The token on place `Mid` is changed from `1‘("NULL",operate)` to `1‘("X1X2",fail)` through the transition `fire`. Afterward, when at least one of the F_events is fixed, the transition `solve` triggers, and the token on the place `Mid` will turn into `1‘("NULL",operate)`.

In CPN models, guards are used to monitoring the input variables. Hence, guards can be used to carry out gate logics, as illustrated in Table 8.1. Here, `#idi rec` extracts the `idi` element from the record `rec`.

Table 8.1: Guards functions in different gates

Transition	Gate	Guards
`fire`	AND	[not (#4 X1) andalso not (#4 X2) andalso (#2 mid)]
	OR	[(not (#4 X1) orelse not (#4 X2)) andalso (#2 mid)]
`solve`	AND	[((#4 X1) orelse (#4 X2)) andalso not (#2 mid)]
	OR	[(#4 X1) andalso (#4 X2) andalso not (#2 mid)]

8.3.4 Subnet of F_events

In an actual system, each case (F_event) can have a maintenance procedure (P_event) [96]. Before a fault is treated or fixed, this case may cause hazards. The corrective maintenance of humans is treated as that: when the dispatchers make errors, it takes a period of time to

carry out the redeployment work. In this section, the condition-based maintenance is not considered in the maintenance components.

"The most commonly used distributions for maintainability analysis have been the normal, lognormal, and exponential," from Handbook, Electronic Reliability Design released by the US Department of Defense [97]. Hence, the exponential distribution is selected. In order to simplify the description, the following assumptions are considered when applying maintenance in this section:

- the downtime required for corrective maintenance time is set to follow an exponential distribution;
- the downtime required for the periodic maintenance is negligible;
- after the maintenance procedure, the F_event is as new as original;
- the failure behavior is stochastically independent.

The P_event of the maintenance process and the chart flow are shown in Fig. 8.6. When the F_event is out of service, the staff is required to carry out the corrective maintenance. Afterwards, the F_event is reset. The periodic maintenance is implemented based on the schedule, and no additional staff is required.

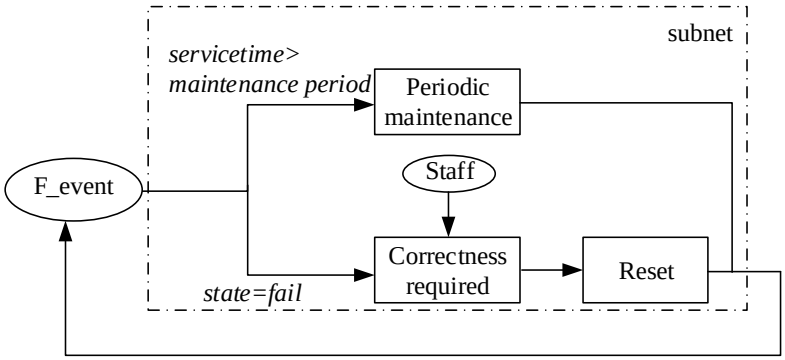


Figure 8.6: Chart flow of the P_event subnet

Fig. 8.7 shows the CPN model of the chart flow in Fig. 8.6. Transitions `Repair_C` and `Repair_P` represent corrective and periodic maintenance, respectively. When a corrective maintenance is required, the staff will be informed immediately. The F_event is in the fault state (place `Failed`) during the corrective maintenance, which is controlled by function

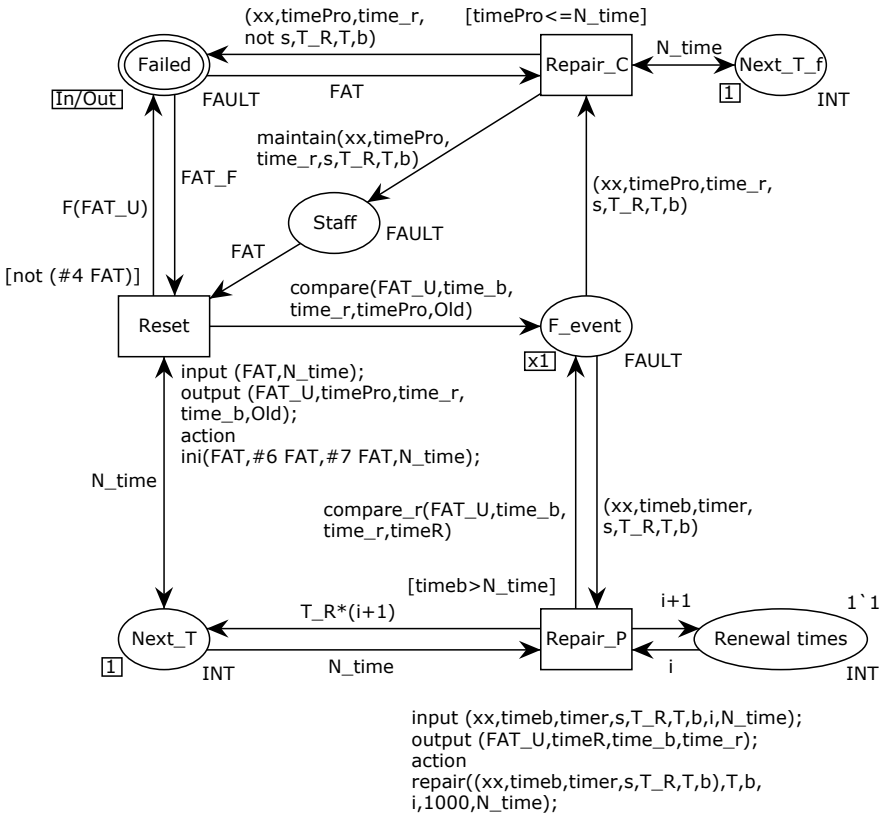


Figure 8.7: Net structure of treatment subnet

`maintain()`. The function adds a maintenance down time to the time stamp. After that, the F_event's state will be initialized by transition `Reset` after the maintenance is completed.

Transition `Repair_P` implements the periodic maintenance, which can be described as a function `repair()` in SML, as shown in Fig. 8.8, where `timeR` indicates the F_event service time after the i th periodic maintenance (`time_r`).

```
fun repair((m,time_b,time_r,s,T_R,T,b):FAULT,
lamd,k,i,Trp,N_time)=
let
val timeR=round(weibull(lamd, k))
val time_r=Trp*(i+1)
val time_b=Trp*i+timeR
in
((m,time_b,time_r,s,T_R,T,b),timeR,time_b,time_r)
end
```

Figure 8.8: Function `repair` in SML

8.3.5 Verification of the new approach for analyzing FT in CPN

In order to verify the accuracy of the new modeling approach, the following two verification (**V**) are considered:

- **V1**: For quantitative analysis, parameterizations and numerical calculations will be implemented in the CPN model. Hence, the accuracy of these calculations has to be verified.
- **V2**: For qualitative analysis, the gate logic is carried out by using SML. Hence, the correctness of the model structure has to be checked.

V1 can be applied to verify the calculation accuracy of the CPN model. The maintenance components are the main numerical operations parts. If F_event `X1` failed, a corrective maintenance is triggered. From the definition of failure probability, it represents the probability that an item will fail during a time interval τ. Equation (8.1) can estimate the failure probability of a F_event with periodic maintenance [95].

$$F(\tau) = 1 - \exp\left[-\left(k \cdot \left(\frac{T_{PM}}{T}\right)^{\beta} + \left(\frac{\tau - k \cdot T_{PM}}{T}\right)^{\beta}\right)\right] \tag{8.1}$$

where T_{PM} is the periodic maintenance period; k is the k th periodic maintenance, $k \in \mathrm{N}$; T is the characteristic lifetime; β is the shape parameter.

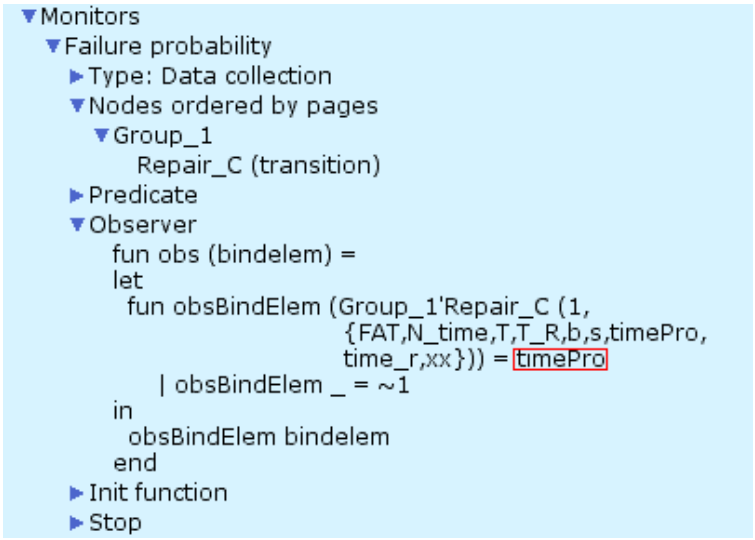


Figure 8.9: Data collection monitor of transition

For simulation, we set the service time τ of F_event `X1` follows a two parametric Weibull distribution with $T = 2000$, $T_{pm} = 1000$ and $\beta = 1.5$. In the CPN model, the failure

probability of `X1` can be calculated by monitoring the transition `Repair_C`, as shown in Fig. 8.9. When the transition `Repair_C` fired, the time stamp will be recorded. In order to obtain the simulation data in the CPN model, the monitor in *CPN Tools* is applied. The monitor is a mechanism that is used to observe and inspect the simulation of a CPN model [34]. We assume that the reader has a basic understanding of it. Otherwise, publication [98] is suggested for starters.

After obtaining the simulation data, we compare it with the formula results that obtained by using the mathematical calculation. The results of **V1** are presented in Fig. 8.10. The cumulative distribution function (CDF) of `X1` during 4000 periods obtained by simulation and theoretical approaches are similar. It indicates that using CPNs can accurately carry out parameterization and numerical calculations.

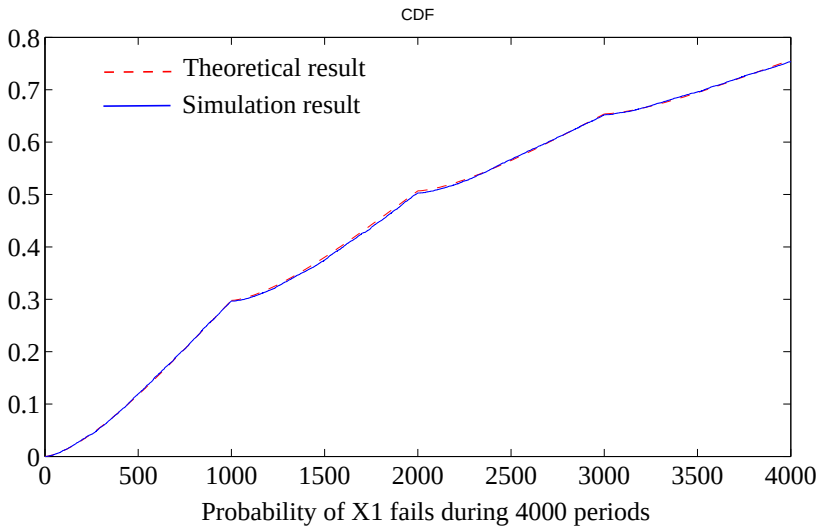


Figure 8.10: Failure probability of the basic F_event with maintenance

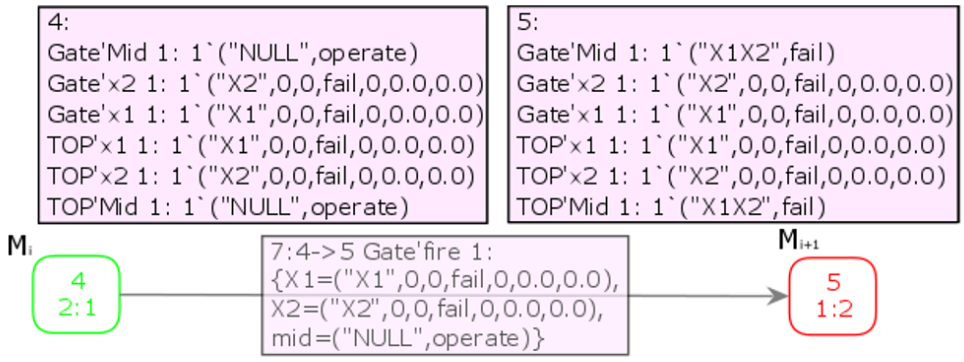
To implement **V2**, the state space analysis is required. Take the AND gate as an example, when `X1` and `X2` happened, the responding F_event `Mid` must occur. Here, M_i represents that both `X1` and `X2` are failed; step Y represents the gate logic; M_{i+1} means the `Mid` is failed. This attribution can be checked by the following query, as shown in Fig. 8.11. Markings M_i and M_{i+1} are found through function `Operating()`. Here, M_i is found in the **node** 4, which represents that both F_events `X1` and `X2` are failed. In addiction, **arc** 7 is the only out **arc** of M_i, and its destination marking is M_{i+1}. The binding **transition** of **arc** 7 is `Gate'fire`. The results mean the AND gate logic meets the requirement.

fun Operating()=PredAllNodes (
fn n=>Mark.TOP'Mid 1 n=[("NULL",operate)]
andalso Mark.TOP'x1 1
n=[("X1",0,0,fail,0,0.0,0.0)]
andalso Mark.TOP'x2 1
n=[("X2",0,0,fail,0,0.0,0.0)]);
val Mi=Operating();
OutArcs(4)
	val Operating= fn : unit->Node list
	val Mi=[4]:Node list
	val it=[7]:Arc list

(1) Relative markings Mi and its out arcs

val Y=ArcToTI(7);
val Mi+1=DestNode(7);
Mark.TOP'Mid 1 Mi1;
Mark.TOP'x1 1 Mi1;
Mark.TOP'x2 1 Mi1;
	val Y=Gate'fire 1:TI.TransInst
	val Mi+1=5:Node
	Val it=[("X1X2",fail)]:INTER ms
	Val it=[("X1",0,0,fail,0,0.0,0.0)]:INTER ms
	Val it=[("X2",0,0,fail,0,0.0,0.0)]:INTER ms

(2) Step Y and markings Mi +1



(3) Representation in state space

Figure 8.11: State space analysis for AND gate logic verification

8.4 New approach for evaluating FT and application to MA+

Based on the translation rules in section 8.3, the collision failure model in Fig. 8.12 can be converted into CPN models, as shown in Fig. 8.13 and Fig. 8.14. The definitions of color set structure, gate structure, and subnet of F_event are in chapters 8.3.2, 8.3.3, and 8.3.4, respectively. Here, over speed and collisions with other obstacles are not included in the discussion. This case focuses on the efficiency evaluation of MA+, so the scenarios related to the movement authority are considered. In the model, the basic F_events are indicated by green symbols. The MA+ is based on the train-centric communication technology, the fundamental failure of the MA+ is the transmission failure.

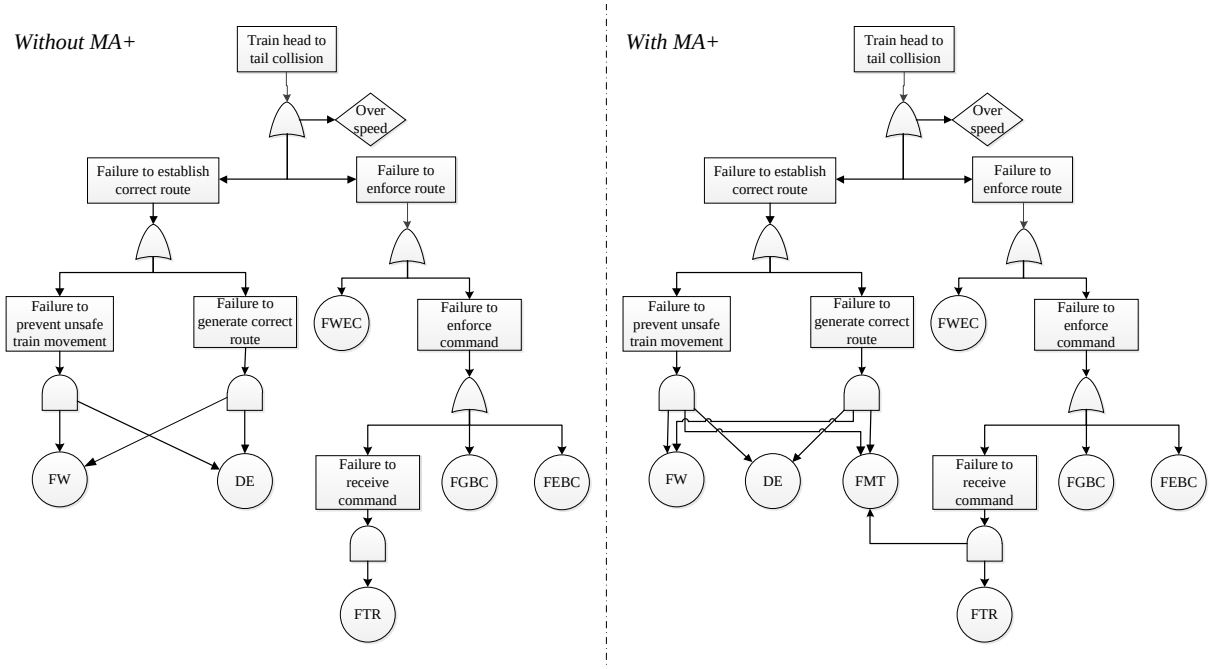


Figure 8.12: Fault Tree of the collision failure model without and with MA+

8.4.1 Qualitative evaluation

Before parameterization, the CPN model correctness should be checked. Here, three most important aspects are evaluated.

- The absence of deadlock markings. Dead marking is a unique marking, which reflects the termination of the system operation and the unavailability of further subsequent actions. However, the railway system is a real-time system, and it has to operate infinitely.

- The absence of self-loop. Self-loop means the system holds on the same marking through an executable transition, and it consumes additional system resources. Self-loop may be a design defect, which should be avoided and corrected.

- Boundedness properties of the top F_event. Boundedness properties represent the token details on a particular place instance. In this chapter, the boundedness properties are used to calculate the minimum cut sets of FT.

To verify the model correctness, a full state space needs to be calculated. In the verification process, no parameters and time attributes are considered, which can avoid the state explosion problem. Under a full state space, all the system variable states are calculated. The

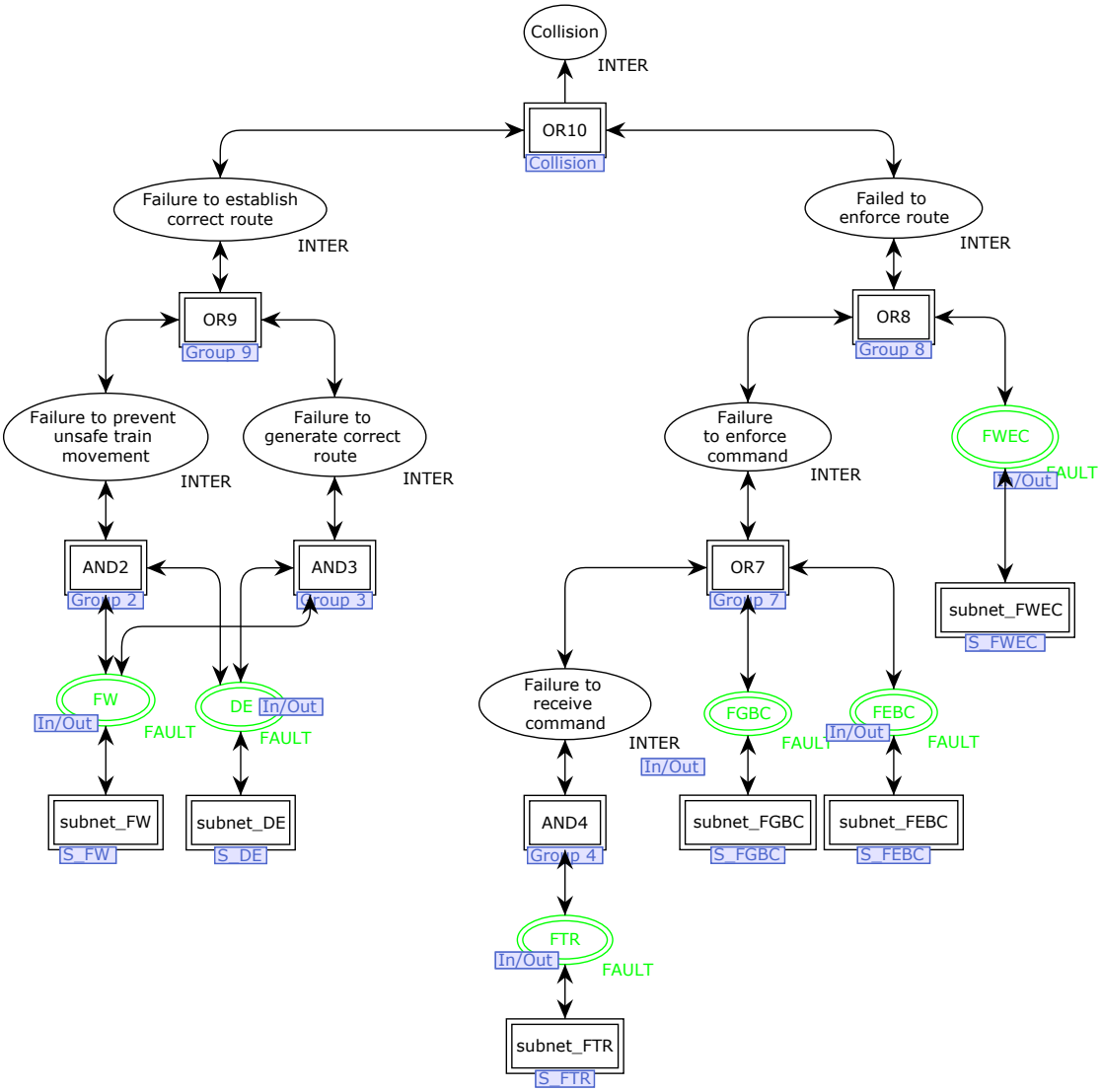


Figure 8.13: CPN representation of collision failure model without MA+

computer used for this calculation is a laptop with a 3.2 GHz Intel i5 processor and 4 GB of RAM. The state space generation results of the CPN model is shown in Fig. 8.15.

There are 138 markings and 199 arcs. The status of state space is `Full`, indicating that all available states have been calculated in the state space.

On the premise of no state explosion, dead marking reflects the abnormal system termination, and self-loop is an indicative of the endless loop in the system. The minimum cut set is for FT qualitative analysis. Fig. 8.16 (1) shows the SML query verifying that the model does not include deadlock markings. By checking the deadlock markings with function `EvalNodes`, the result demonstrates the absence of deadlock markings. All the F_events discussed here have maintenance P_event, and no deadlock markings are permitted in this model.

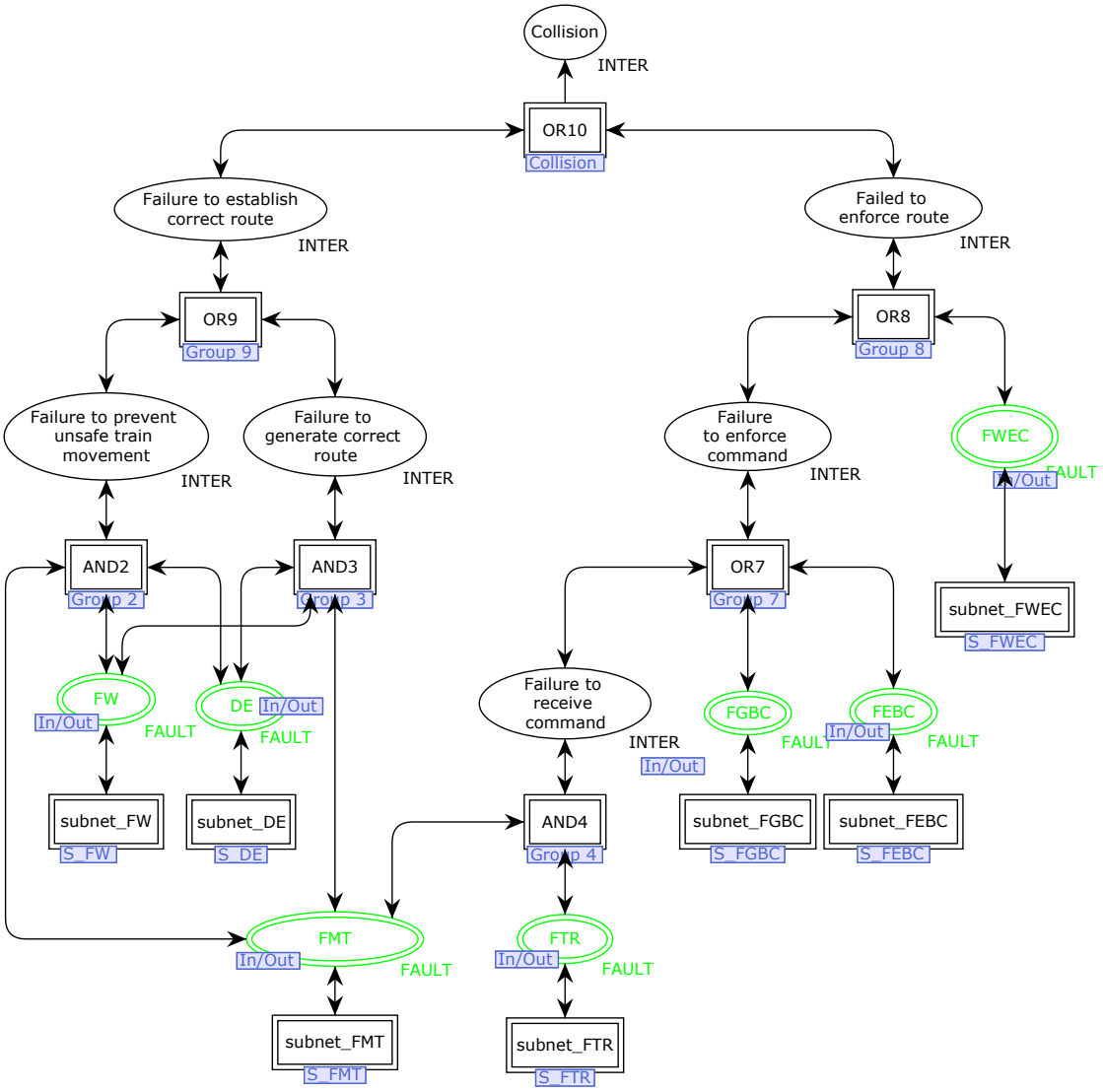


Figure 8.14: CPN representation of collision failure model with MA+

The query that demonstrates the absence of self-loop is shown in Fig. 8.16 (2), function `SelfLoopTerminal` checks whether the marking will hold the same marking. The query results in Fig. 8.16 (2) confirms the correctness of the CPN model.

The boundedness properties response to the token details on a specific place instance. In the Fig. 8.14, both the intermediate F_events and top F_event have color set `INTER`. It will record which basis F_events triggered the intermediate F_events and top F_event. Hence, the boundedness properties provide a new qualitative analysis approach to study the FT minimum cutsets. Using prime number method, binary bit string method can easily carry out the minimum cut-set [99]. The SML query for the minimum cut-set could be specified as in Fig. 8.16 (3).

Statistics

State Space
Nodes: 138
Arcs: 199
Secs: 0
Status: Full

Figure 8.15: State space statistics report (partical)

Checking queries	Results
let val fid = TextIO.openOut "Dead Marking results.txt" val _ = TextIO.output(fid, "List of dead markings: \n") val _ = EvalNodes(ListDeadMarkings(), fn n => INT.output(fid,n)) val _ = TextIO.output(fid, "\nNumber of dead markings: ") val _ = INT.output(fid,length (ListDeadMarkings())) in TextIO.closeOut(fid) end	List of dead markings: Number of dead markings: 0
(1) Absence of dead markings	
fun SelfLoopTerminal n=(OutNodes(n)=[n]) fun InValidTerminal()=PredNodes(EntireGraph, fn n => (SelfLoopTerminal n), NoLimit); let val fid = TextIO.openOut "ListOfSelfLoops.txt" val _ = if InValidTerminal()=[] then TextIO.output(fid, "There is no self loop terminal!\ n") else TextIO.output(fid, "List of self loop terminals: \n") val _ = EvalNodes(InValidTerminal(), fn n => INT.output(fid,n)) in TextIO.closeOut(fid) end	There is no self loop terminal!
(2) Absence of self-loop	
let val fid = TextIO.openOut "Minimum cut-set.txt" val _ = TextIO.output(fid, "\nMinimum cut-set: ") val _ =TextIO.output(fid, STRING.mkstr_ms(UpperMultiSet(Mark.TOP'collision 1))) in TextIO.closeOut(fid) end	Minimum cut-set: 1`("",operate)++ 1`("++FWEC",fail)++ 1`("+FEBC+FGBC++FWEC",fail)++ 1`("+FEBC+FGBC+FTFMT+FWEC",fail)++ 1`("+FMTDE+",fail)++ 1`("FEFMT++",fail)
(3) Minimum cut-set	

Figure 8.16: (1) Absence of dead markings,(2) absence of self-loop,(3) minimum cut-set

8.4.2 Parameterization procedure

In this section, database reports and statistical data are applied to carry out the parameterization procedure, which estimates the parameters of F_events' service time. Note that these assumed parameters and distributions are not real parameters of the railway system. They are only used for illustrating the performance of our approach.

For the F_events that have practical data, the parameters are calculated based on the Safety Database Activity Report Significant Accidents from 2011-2014. Moreover, the numbers of accidents were almost the same in each year from 2011-2014. Hence, we assumed that the successful service time of basic F_event A is τ_A. F_event A failed N times during the observation time T_0. Hence,

$$\tau_A = \frac{T_0}{N} \quad h \tag{8.2}$$

For an exponential distribution, its expectation is

$$\tau_A = \frac{1}{\lambda} \quad h^{-1} \tag{8.3}$$

$$\lambda = \frac{N}{T_0} \quad h^{-1} \tag{8.4}$$

where, λ is the parameter needs to be calculated.

Table 8.2: Database reports and statistical data of F_events

Database reports from 2011 to 2014			
Causes	Number of accidents	Observation time T_0 (h)	Relative F_events
Human factors	125	35040	`DE`
Rolling stock	5	35040	`FGBC, FEBC`
Control-command systems	0.2% of accidents	35040	`FW, FWEC`
Statistical data of systems			
Systems	Working time (h)	Failure possibility	Relative F_events
MA+ transition failed	> 200	< 0.2	`FMT`
GSM-R communication failed	> 1	$< 10^{-4}$	`FTR`

We collected the number of significant accidents of the train to train collisions in Europe, the results are shown in Table 8.2. Among causes of train to train collisions, human factors represented the first cause of accidents. The observation time T_0 is 35040 hours, $N = 125$.

Based on the equation 8.8, the distribution of DE can be assumed as an exponential with a parameter λ_{DE}. Hence,

$$\tau_{DE} = \frac{T_0}{125} = \frac{1}{\lambda_{DE}} = 280 \quad h \tag{8.5}$$

$$\lambda_{DE} = 3.57 \times 10^{-3} \quad h^{-1} \tag{8.6}$$

Similarly, there were 5 train to train collisions caused by rolling stock. We assumed that fault F_events FGBC and FEBC related to onboard equipment have the distribution parameter of rolling stocks $\lambda = 4.75 \times 10^{-5} h^{-1}$. The existing train signal system has a high-safety level, the failure of control-command signaling and operations & traffic management systems caused 0.2% of accidents [42]. Here, we assume the failure distribution of FW and FWEC with an expectation working time of 2190 h, the parameter $\lambda = 4.56 \times 10^{-4} h^{-1}$.

For certain systems or technologies, practically statistical data is possible. A system service time τ follows an exponential distribution [100]. The system can be assumed to work more than x hours in P of all cases. Hence,

$$P = e^{-\lambda x} \tag{8.7}$$

$$\lambda = -\frac{\ln(P)}{x} \quad h^{-1} \tag{8.8}$$

For the FMT, it is supposed to work 200 hours without fault in 80% of all the cases. Based on the equation (8.7) and (8.8), the distribution parameter is: $\lambda_{FMT} = 1.12 \times 10^{-3} h^{-1}$. For the FTR in the GSM-R communication, a complete connection loss takes place 10^{-4} times per hour [76]. The parameter of the FTR is set accordingly. The distribution parameters of basic F_events are shown in Table 8.3.

Table 8.3: Places corresponding meaning and distribution parameter in the CPN model

Elements	Corresponding meaning	$\lambda(h^{-1})$
FW	Failure of wayside equipment	4.56×10^{-4}
DE	Dispatcher error	3.57×10^{-3}
FMT	MA+ transmission failed	1.12×10^{-3}
FTR	Transmission failed	1.00×10^{-4}
FGBC	Failed to generate braking commands	4.75×10^{-5}
FEBC	Failed to execute braking commands	4.75×10^{-5}
FWEC	Failure of wayside equipment to execute the correct commands to protect against conflicting routes	4.56×10^{-4}

8.4.3 Dependability analysis

With the assistant of monitors and simulation report [4], all the data details are available to do the analysis. In this section, the failure probability of the top F_event is illustrated as a case study.

The unreliability, $F(\tau)$, a measure of failure, is defined as the probability that a system is failed under given conditions for a given time interval τ [41]. Hence,

$$F(\tau) = P(TE \leq \tau) \tag{8.9}$$

where TE indicates the top F_event, i.e. the first time the collision occurs after the last maintenance; $P(TE \leq \tau)$ is the system unreliability, i.e. the cumulative probability function of TE until τ. Hence, the probability of failure is defined as:

$$f(\tau) = \frac{F(\tau)}{d\tau} \tag{8.10}$$

For 1.0×10^8 period simulations, the simulation report frame is shown in Fig. 8.17. In the report, one period represents one hour. The report illustrates when a transition will fire, and the corresponding binding elements are given. As shown in step 109, one collision occurs at the 3946 periods, and the hazardous F_events cause this collision are `FWEC`. The fault `FWEC` is treated at the 3953 periods, the collision problem solved at the same time.

With the simulation results, the cumulative failure probability of the F_event `collision` can be calculated. All the time interval data is divided into 50 parts, and the median values

[4] A simulation report is a text file that contains information regarding the transitions that occur during a simulation.

```
109 3946 fire @ (1:Collision)
- M1 = (“”,operate)
- M2 = (“+FWEC",fail)
- mid1 = ("++FWEC",fail)
110 3953 Reset @ (1:S_FWEC)
- FAT_F = ("FWEC",3946,4000,fail,1000,2190.0,1.0)
111 3953 solve @ (1:Group_8)
- M1 = (“”,operate)
- X2 = (“”,4105,4000,operate,1000,2190.0,1.0)
- mid = (“+FWEC",fail)
112 3953 solve @ (1:Collision)
- M1 = (“”,operate)
- M2 = (“”,operate)
- mid1 = (“”,operate)
```

Figure 8.17: Timed CPN simulation report (partial)

are used to do data fitting. As shown in Fig. 8.18 and 8.19, the results indicate that the cumulative failure probability of collision has a reduction when MA+ is applied.

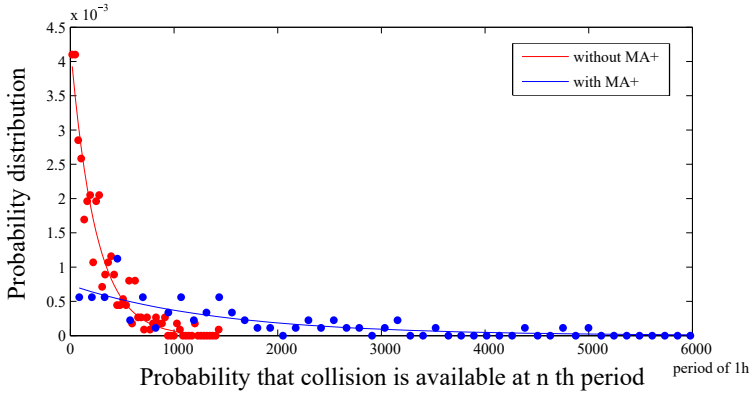


Figure 8.18: Probability density function of the collision happens at time t

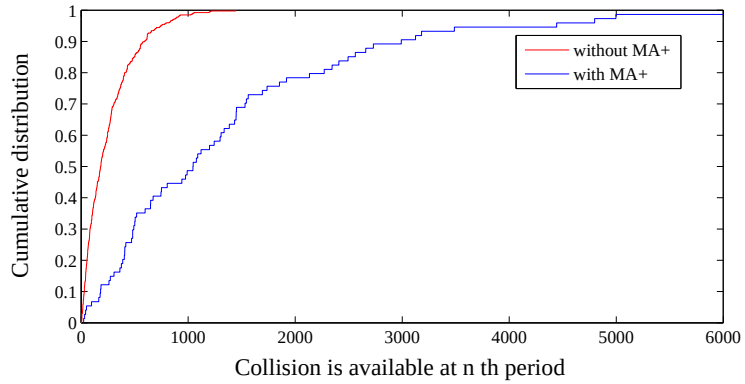


Figure 8.19: Cumulative distribution function of the collision happens at time t

Based on the above case study, the new approach is more powerful to analyze the dependability of a complex system such as railway systems.

8.5 Summary

This chapter introduced an approach to transfer and evaluate the regular FT in CPN models for the system dependability analysis. The approach can be reused and re-customize, which is generally limited to other FT commercial products. With the assistant of CPNs, the evaluation process can carry out the F_event subnet operations, model correctness verification, time factors, mathematical calculation, and so on. In detail, the approach permitted application of different fault rates and consideration of maintenance components. All the F_events and gate behavior were treated as duration time conditions.

In the verification approach, both quantitative and qualitative analyses were accomplished in the CPN model. The accuracy of the verification was demonstrated by comparing the simulation and mathematical calculation data. The model's correctness is verified by means of state space analysis. The evaluation results proved that more information was available than FT to do the failure rate and availability analysis, and the MA+ can reduce the probability of collision.

In further works, more gates could be considered for different conversion rules. One can propose a better organized and thorough checking process in the formal model analysis and introduce different checking queries more systematically. Furthermore, as soon as the distributions of basic F_events are available, one can analyze experimental data and apply them to the CPN model for dependability assessments.

9 Practical implementation in the metro

In this chapter, we apply the TTDMS in the metro system. After introduced the system strategy, the stochastic Petri nets model is applied to assess the system availability. Additionally, in order to evaluate the system's functional performance, it is deployed in the real world. The numerical validations are implemented. Both mathematical calculation and software simulation are considered. The system performance is validated in distance detection and measurement accuracy. The detection range is verified in straight and curved lines through computer simulation and mathematical calculation. The TTDMS detection distance is simulated by using Wireless InSite®, and the simulation results indicate that the method applied in TTDMS is feasible.

Additionally, a TTDMS prototype machine is assembled by Tongji University (China). More importantly, experimental data is collected from a metro line in use. Static and dynamic measurements are considered in both tunnel and viaduct scenarios. With the practical measurement results obtained by the prototype machine, the distance measurement accuracy is evaluated. Based on the results, the train to train distance is proved to follow a Gaussian distribution, and TTDMS can satisfy a daily metro train's operation in terms of the distance requirement. By comparing the obtained data with the record of Automatic Train Supervision (ATS), the result that TTDMS is feasible in real metro scenarios is achieved.

The chapter is arranged as follows: Section 9.1 describes the TTDMS in the metro system, which is a speed based distance warning strategy. What is more, the distance measurement unit and distance calculation flowchart are introduced. Section 9.2 evaluated the system availability by using the stochastic Petri net. The prototype machine is presented in section 9.3.

Sections 9.4 and 9.5 are seeking to verify TTDMS's performance through these three steps:

- Mathematical calculation and simulation results are used to estimate the detection range. The simulation results confirm that TTDMS is feasible in actual railway engineering applications.

- With the experimental results obtained by our prototype machine, the distance measurement accuracy is evaluated. Comparing TTDMS to the data in Automatic Train Supervision (ATS) shows that TTDMS is feasible in real metro scenarios.
- Model-based execution time estimation is used to do the static analysis of the software implementation. With proper parameters, the system performance can be evaluated.

Finally, Section 9.6 concludes this chapter and presents future works.

9.1 TTDMS in metro

A metro is a typical high-capacity public transport in metropolitan areas. It constitutes a considerable part of the public transport. Different from transportation on the ground, such as buses and trams, the metro system is localized in tunnels or on viaducts. In these scenarios, the metro system cannot be accessed by any other vehicles or pedestrians. To ensure a safe operation, the metro train control system is subject to strict fault-to-safe regulations. When a train is under the supervision of a metro control system, the risk is limited. However, a loss in efficiency of the signal system or the ATP equipment leads to trains running without supervision. For example, when the communication between trains and the ground control center fails, or when the ATP needs to be deliberately disabled for a high operation efficiency. Once the train to ground connection is interrupted, the probability of train collision will increase. Hence, it is essential to do some research in the domain of train collision avoidance. Developing a distance measurement system that can be applied in various scenarios, especially in the underground is the most important issue.

9.1.1 Speed based distance warning strategy

In the close proximity driving, the braking distance relates to the train speed. As shown in Fig. 9.1, we divided the train distance interval into several sections. S_w is the warning distance period, which will provide 6 s for warning information for the train driver before a braking action is required. $S(v)$ is the braking distance. Since the TTDMS is implemented in the STM32F205RET6 prototype machine, a safe and efficient braking distance calculation strategy is proposed, as shown in Eq. (9.1).

$$S(v) = S_{de} + S_{tr}(v) + S_{eb}(v) + S_{sp} \quad m \tag{9.1}$$

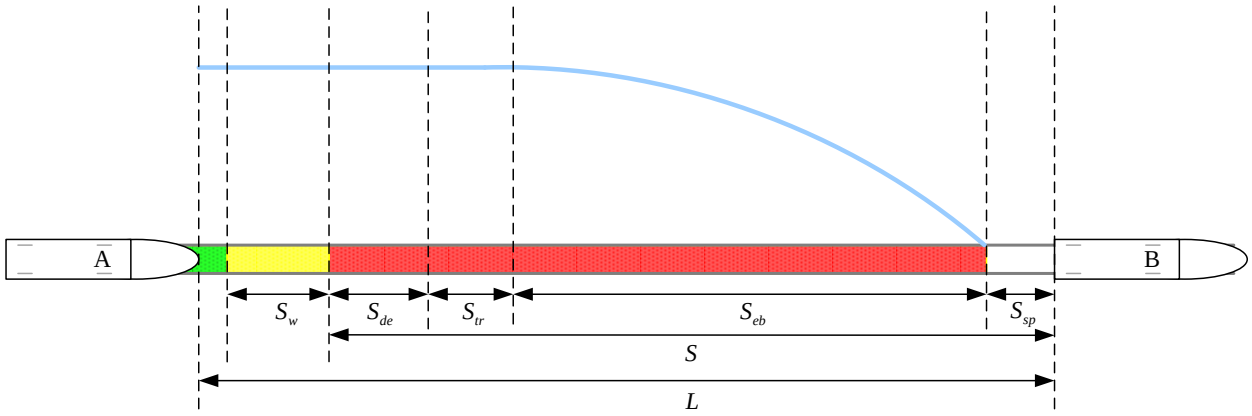


Figure 9.1: Braking procedure

where S_{de} is the distance measurement error caused by the Non-line-of-sight (NLOS) and calculation; $S_{tr}(v)$ is the train movement distance in the reaction delay; $S_{eb}(v)$ is the braking distance; S_{sp} is the safety prevent distance.

S_{de} is related to the measurement distance L, as shown in Eq. (9.2). The parameter β can be obtained based on the actual measurement results; S_{sys} is the system error.

$$S_{de} = \beta \cdot L + S_{sys} \quad m \tag{9.2}$$

$S_{tr}(v)$ is given by Eq. (9.3).

$$S_{tr}(v) = (1 + \delta_e) \cdot v \cdot t_{tr} + \frac{1}{2}(a_{tra} + a_r + a_g) \cdot t_{tr}^2 \quad m \tag{9.3}$$

$$\left| \frac{v_r - v}{v} \right| \leq \delta_e \tag{9.4}$$

where δ_e is the error coefficient of speed measurement, and it is defined by Eq. (9.4); v is the measured train speed and v_r is the actual train speed, respectively. Due to the error in speed measurement, we choose the worst case for the initial speed in the train braking procedure, which is the maximum possible speed according to the speed measurement. t_{tr} is the time consumed to establish the braking action; a_{tra} is the current traction acceleration; a_r is the resistance acceleration usually caused by rolling friction and air resistance; a_g is the current line slope, curve or tunnel acceleration.

In details, the characteristics of a_r is shown as a traditional quadratic formula the so-called

Davis formula [101] in Eq. (9.5), and a_r is shown in Fig. 9.2. Those accelerations are changing with the train speed, but we can regard them as constants for the efficacy of calculation. Additionally, due to the time t_{tr} of the braking establishing is extremely short, the train speed will not increase too much.

$$a_r = A + B \cdot v + C \cdot v^2 \tag{9.5}$$

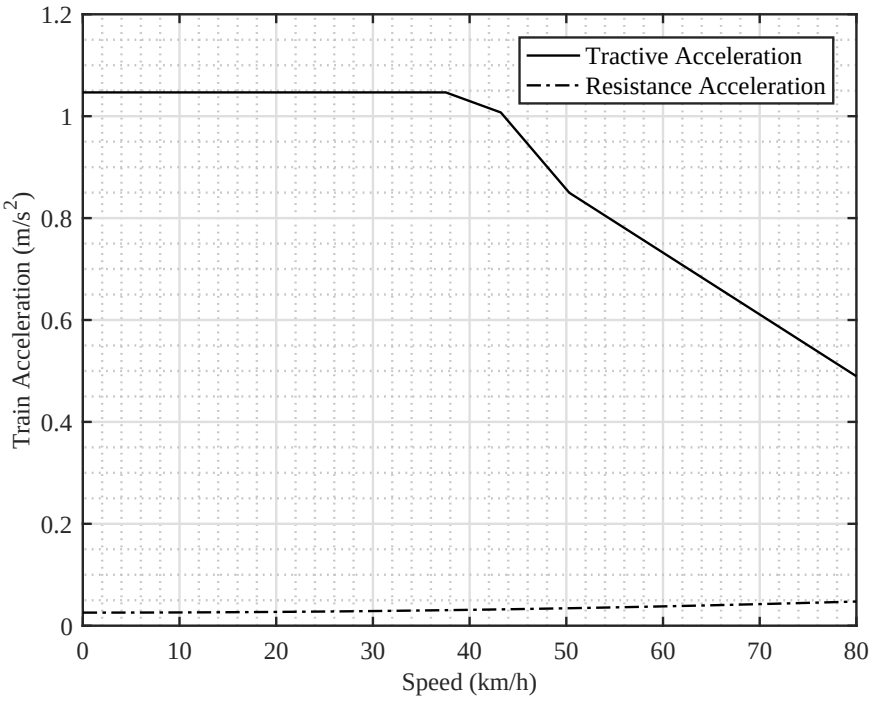


Figure 9.2: A typical graph of tractioin/resistance acceleration with no adhesion

S_{eb} is related to the start and end speeds, and it is given by:

$$S_{eb}(v) = \frac{v_b^2 - v_e^2}{2a_{eb}} \quad m \tag{9.6}$$

where v_b is the start speed of the braking procedure, also it is the end speed of the reaction delay procedure for the continuity of train speed and can be obtained in Eq. (9.7); v_e is the end speed and a_{eb} is the braking deceleration which is regarded as a constant here.

$$v_b = (1 + \delta_e) \cdot v + (a_{tra} + a_r + a_g) \cdot t_{tr} \quad m/sec \tag{9.7}$$

9.1.2 Distance measurement unit

The TTDMS can obtain the extra information by applying the train-centric communication. The additional information includes but not limits to other trains' speed and position. The structure of distance measurement core unit is shown in Fig. 9.3. The digital map stores the line data [102].

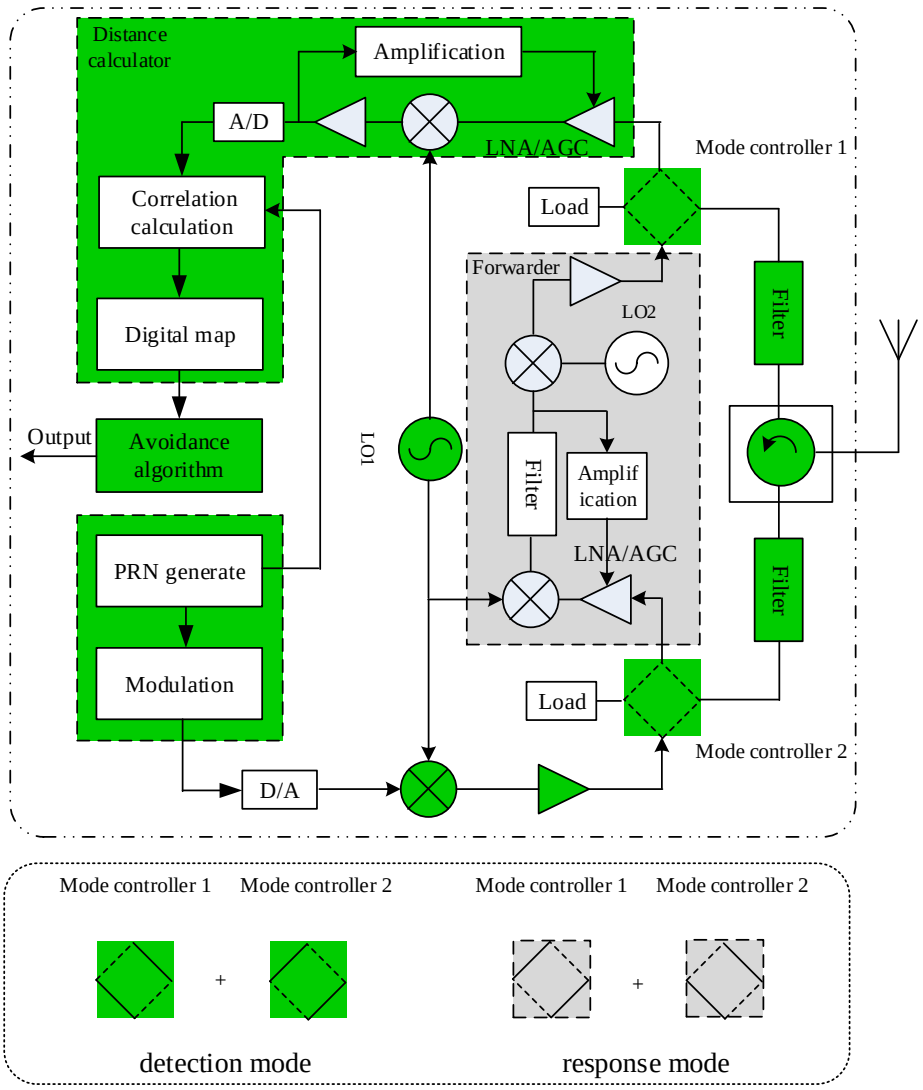


Figure 9.3: The distance measurement core unit

This unit involves six parts: transceiver, distance calculator, avoidance algorithm, PRN generator, forwarder, and the mode controller. The transceiver is in charge of exchanging data. The distance calculator does the correlation calculation and matches the distance information with the digital map. The avoidance algorithm analyzes the distance and speed

information, then outputs appropriate commands. The PRN generator generates PRN code. The forwarder resends the received signal. The mode controller makes both following and detected trains share the same system architecture, but carry out different functions. For example, when the system is in the detection mode, as shown in Fig. 9.3 the forwarder section that marked with the gray background is disabled; the distance calculator, avoidance algorithm, and PRN generator that identified with the green background are implementing the detection function.

The distance measurement flowchart is shown in Fig. 9.4. In the detection mode, the train sends signal $s(t)$ involving PRN code sequence; when the train obtains the forwarded signal $r(t)$, the system implements distance measurement calculation and outputs results. In the response mode, the train will forward the received signal to the following train.

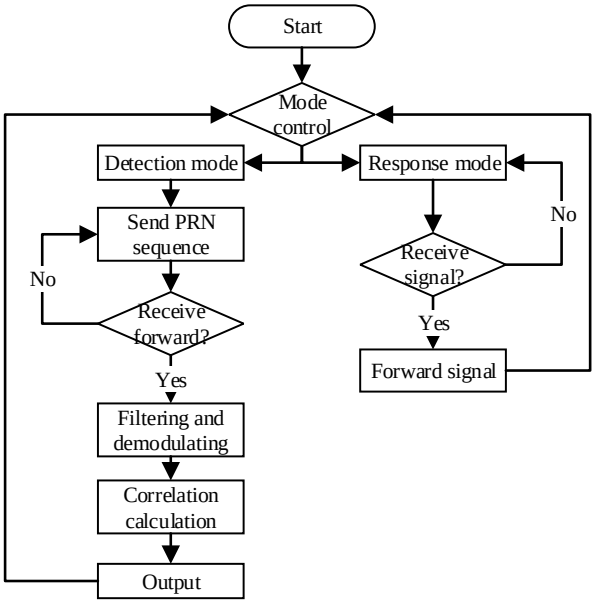


Figure 9.4: Distance calculation flowchart

9.2 System availability evaluation

In this section, a quantitative model of the system availability is presented and analyzed. The strategy for processing the distance results is taken into account. The quality of communication architecture is given in the specification of the Euroradio [103]. The model is based on the specification to do the parameterization. We assumed that readers have basic knowledge of stochastic Petri nets.

In order to evaluate the system availability, stochastic Petri nets are applied [76], as shown in Fig. 9.5. In the model, the system states represented by places, and system functions are indicated by transitions. There are three different arcs in the model, they are normal, test, and inhibitor arcs, respectively [104]. Normal arcs allow the modifications of states by transferring tokens among places through transitions. Test arcs have the same function as normal arcs, but not lead to consuming the tokens. Inhibitor arcs make sure that a place must be empty for the transition to get activated. During the simulation, one unit of the model time means on second in reality. The meanings of places are shown in Table 9.1.

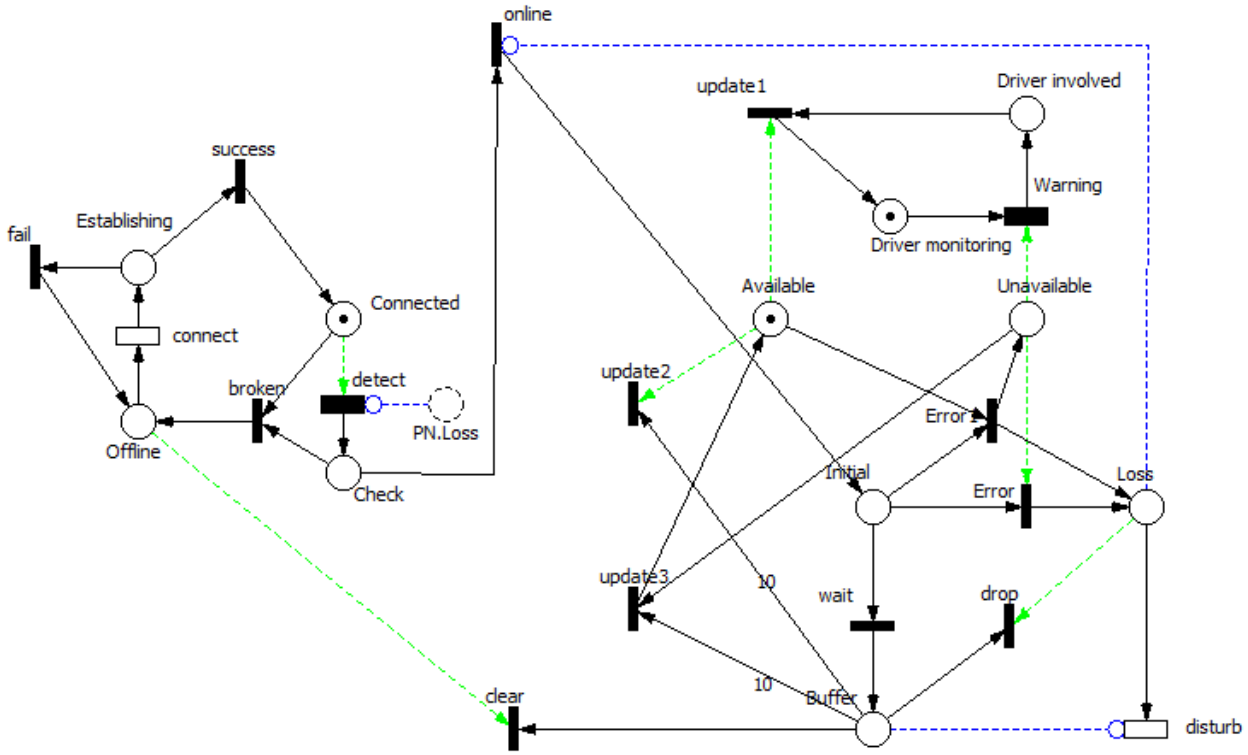


Figure 9.5: Availability evaluation

The distance measurement strategy is proposed as follows: ten times continuous initial measurement results will be stored in the buffer; based on these ten initial data, the system will output a smoothing result through transitions `update1`, `update2`, and `update3`. When there is no smoothing distance result is available, the token will transfer from `Available` to `Unavailable`. Additionally, if the TTDMS is unavailable for more than 6 s, which is modeled by transition `Warning`, the driver has to be involved to maintain the operation safety until the system is available again.

The system carries out the initial distance measurement every 50 ms. The connection loss happens $< 10^{-2}$ per h or $< 2.77 * 10^{-6}$ per second. Hence, transitions `broken` and `online` are

set with possibilities $< 2.77*10^{-6}$ and $1-2.77*10^{-6}$, respectively. When the system is offline, transition `connect` models the connection establishment procedure. The time assumed by this procedure is random, but requires to be < 8.5 sec (95%) and follows an exponential distribution. The density and distribution functions of the exponential distribution are given by equations (9.8) and (9.9), respectively.

$$f(x) = \lambda e^{-\lambda x} \tag{9.8}$$

$$F(x) = 1 - e^{-\lambda x} \tag{9.9}$$

Hence, the parameter λ value can be calculated as:

$$\lambda = -\frac{\ln(1-p)}{x} \quad sec^{-1} \tag{9.10}$$

where p is 0.95, x is 8.5 sec, hence $\lambda = 0.35$.

After the establishment, the requirements specify that the connection will be successfully established with a possibility 99%. Hence, the transitions `success` and `fail` are set as deterministic transitions, and with weights 0.99 and 0.01, respectively.

During the data communication process, the transmission problem occurs since the electromagnetic interference will influence the measurement result. It can be treated as the relationship between the bit error rate (BER) and carrier-to-noise ratio (CNR). In our system, the digital modulation process is Binary Phase Shift Keying (BPSK), and the Spreading Gain is 20 dB. Hence, when CNR > 0 dB the BER $\ll 10^{-6}$ [105].

Here we choose the failure rate of distance measurement to be 10^{-4}, which is a more critical value than it in the actual scenario. The parameters of the deterministic transitions `Error` and `wait` are set accordingly. Transition `disturb` models the time delay caused by transmission problem. The delay is assumed to be < 0.8 sec for 95% of all cases. Hence, the parameter $\lambda = 3.7\ sec^{-1}$.

For numerical analysis of the model, Monte Carlo Simulation is applied. The model is simulated with a Windows 10 Pro system, Intel Core i5 3.20 GHz processor, and 4 GB RAM. The computation lasts 7.5 hours to do a simulation duration about 18566 days, equal to 50 train years, in the model. The simulation results are shown in Table 9.1. The system

is working available with a probability of 99.6937%. The probability that the driver has to be involved is $7.22667 * 10^{-8}$.

Table 9.1: Place occupancy probability

Place	Probability	Meaning
Loss	0.00106693	Data loss
Buffer	0.898805	Buffer
Available	0.996937	Distance is available
Unavailable	0.00306337	Distance is unavailable
Driver involved	$7.22667*10^{-8}$	Driver has to be involved
Driver monitoring	1	Driver monitors the equipment
Check	$< 6.20551*10^{-10}$	System self-test
Establishing	$< 6.20551*10^{-10}$	Establishing the connection
Connected	0.99984	System connected
Offline	0.000159658	System offline

9.3 Prototype machine

A prototype machine is designed to carry out the aforementioned TTDMS functions, as shown in Fig. 9.6. The practical measurements are shown in Table 9.2, average errors β are obtained. In the close proximity driving scenario, the delay caused by the braking establishing and driver reaction is set to 1.5 s [106]; the braking deceleration a_{eb} is 0.8 m/s^2; the safety prevent distance S_{sp} is 30 m. Based on the distance warning strategy, Fig. 9.7 shows the relationship among measurement distance, speed, and braking distance.

Table 9.2: Measurement errors

Distance L (m)	Number of measurements	Average errors (%) β
> 800 m	106	14.46
$600 - 800$ m	214	11.98
$300 - 600$ m	55	10.34
< 300 m	28	9.01

In order to assist the driver with essential information just in cases of imminent danger, an easy and simple train-driver interface is introduced, as shown in Fig. 9.8. Symbols or shapes having a certain meaning in general railway control systems are avoided, and the interface permits the no interaction with the system during the normal operation. Both acoustic and graphic alarms to remind the driver when the train is in the S_w section. Additionally, a

Figure 9.6: Prototype TTDMS system in the metro train

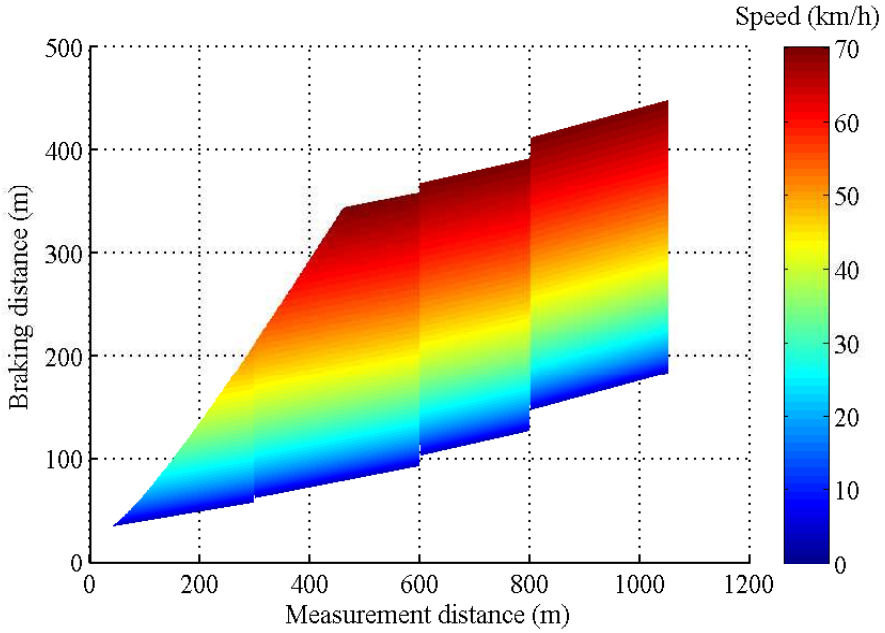


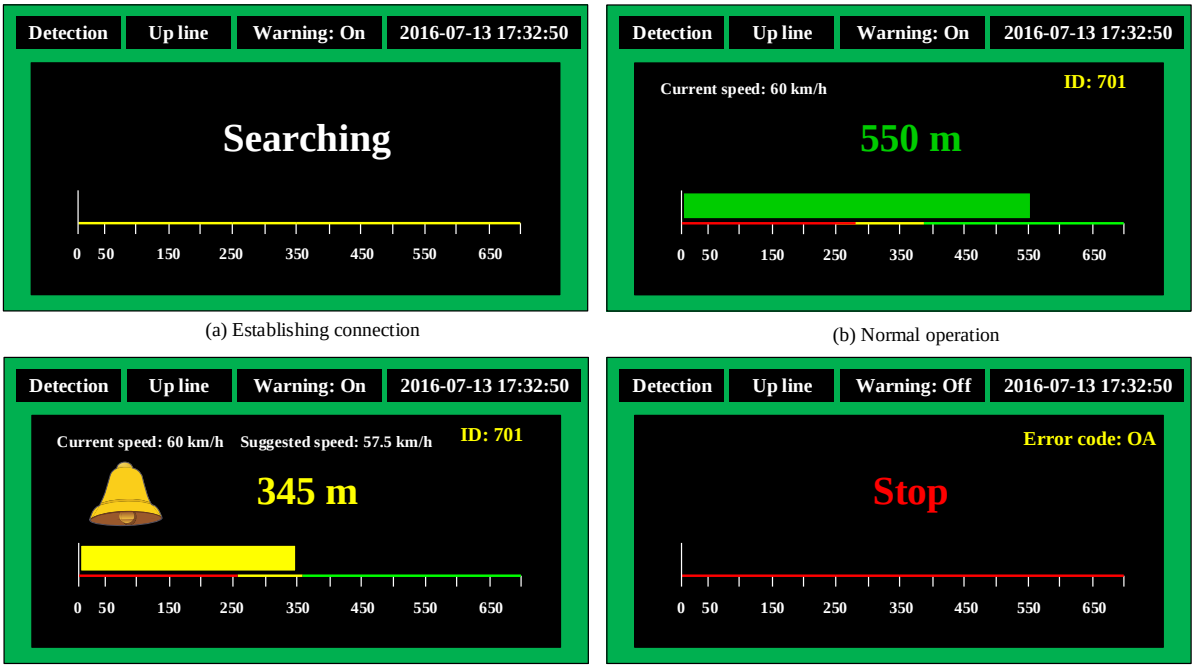
Figure 9.7: Intercorrelations among measurement distance, speed, and braking distance

suggested speed is proposed based the train distance interval. When the system is out of service, an error prompt is available.

In close proximity driving, the train tracking interval limits the transport capacity and efficiency. In order to simulate the efficiency of this system, we simulate the two train tracking procedure based on the actual line parameters of Shanghai metro line 7. The

Table 9.3: Traction calculation

Train ID	From	To	Block m	Departure time (s)	Arriving time (s)
1001	Meilan Lake	Luonan Xincun	1639	0	126
1001	Luonan Xincun	Panguang Road	3040	157	373
1001	Panguang Road	Liuhang	966	403	483
1001	Liuhang	Gucun Park	1754	514	647
1001	Gucun Park	Qihua Road	2534	678	863
1001	Qihua Road	Shanghai University	1691	894	Null
1003	Meilan Lake	Luonan Xincun	1639	67	192
1003	Luonan Xincun	Panguang Road	3040	223	438
1003	Panguang Road	Liuhang	966	468	550
1003	Liuhang	Gucun Park	1754	580	713
1003	Gucun Park	Qihua Road	2534	744	929
1003	Qihua Road	Shanghai University	1691	960	Null



(a) Establishing connection

(b) Normal operation

(c) Warning

(d) Error prompt information

Figure 9.8: The train-driver's interface

minimum distance between two trains is based on the results in Fig. 9.7. When the train arrives at stations, the train will stop for 30 seconds. The details of the traction calculation are shown in Table 9.3.

The simulation results are shown in Fig. 9.9. It indicates that the tracking interval is around 67 ± 2 s, and no braking action is fired due to the safety distance interval.

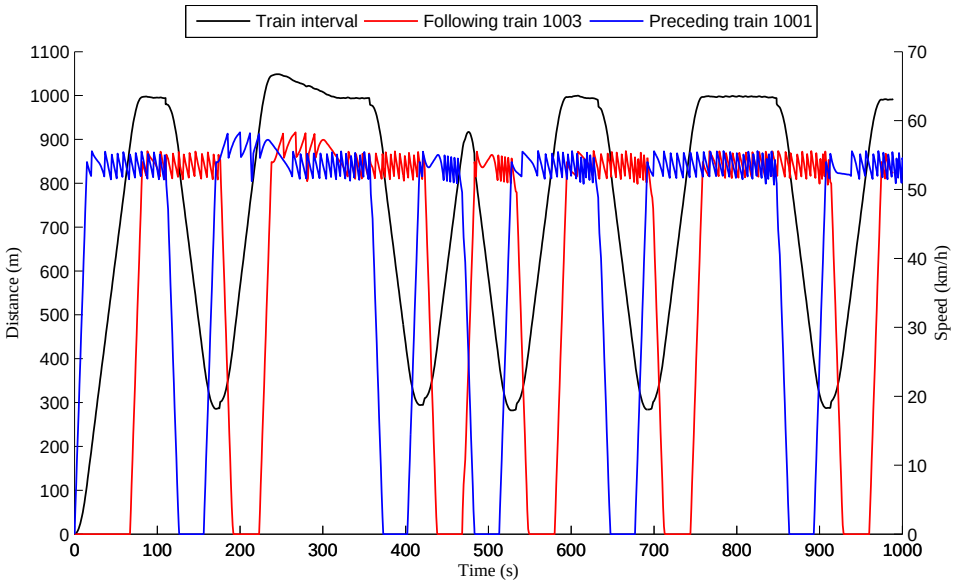


Figure 9.9: Simulation train interval

9.4 Detection distance validation in simulation

The detection distance depends on the hardware ability. A larger detection distance results in a higher transmission power. The collision alarm should be given before the avoidance of a collision becomes impossible. Hence, the detection distance should be further than the braking distance. In the case of a metro train with a weight of 90 t, the braking distance is 318.5 m when the train is moving at an initial speed of 80 km/h [107]. The path loss ($L_B\,[dB]$) limits the distance measurement (d_{Km}), as shown in equation 9.11 [108].

$$P_r\,(d)\,[dBm] = P_t\,[dBm] + G - L_B\,[dB] - PL_\Delta - PL_w \geq sensitivity \tag{9.11}$$

$$G = 10\log\left(G_t G_r\right) \tag{9.12}$$

where $P_r\,(d)\,[dBm]$ is the receive power, $P_t\,[dBm]$ is the transfer power, G is the gain, G_t and G_r are the gain of the transfer and receiver antenna, respectively. $L_B\,[dB]$ is the path loss, PL_w is the attenuation caused by glass windows. $P_r\,(d)\,[dBm]$ should be greater than the sensitivity of the receiver. In an ideal case of no obstacles, the path loss would be the free space loss. The path loss $L_B\,[dB]$ can be calculated using equation 9.13 [108].

$$L_B\,[dB] = 32.44 + 20\log_{10} f_{MHz} + 20\log_{10} d_{Km} \tag{9.13}$$

where f_{MHz} is the frequency of the carrier wave.

To estimate the detection distance of TTDMS, the following specifications are made. The input power is 20 dBm, the maximum gain is 6 dBi, the receiver threshold is -100 dBm, the working frequency of the carrier wave is 2406 $\sim$ 2476 MHz, and the center frequency (2440 MHz) is applied, in the following calculations. The device attenuation $PL_{\Delta} = 0$, the attenuation caused by the glass windows is $PL_w = 6$. Hence,

$$-100 < 20 + 6 - L_B\,[dB] - 0 - PL_w \tag{9.14}$$

$$L_B\,[dB] < 120 \tag{9.15}$$

From equation (9.13) results,

$$d_{Km} = 10^{\left(\frac{L_B\,[dB] - 32.44 - 20\log_{10}2440}{20}\right)} \quad km \tag{9.16}$$

$$d_{Km} \approx 9.77 \quad km \tag{9.17}$$

The maximum propagation distance is 9.77 km in the ideal case of no obstacles. Tunnel and curve lines are common in metro lines. However, the path loss calculations in these scenarios are hard to carry out because the propagation calculations are complex. In order to evaluate the propagation ability in these scenarios, the simulation software Wireless InSite® is applied. The software is a suite of ray-tracing models for the analysis of radio propagation and wireless communication systems. The software's efficient and accurate predictions can be used to undertake our simulation.

The simulation environment is divided into two main parts. The path loss is estimated in a straight line and a curve line. The object model is shown in Fig. 9.10. The tunnel's diameter is 5.5 m, the curve line has a length of 550 m with a radius of 360 m. The straight tunnel line is 1653 m. The simulation parameters' settings are shown in Table 9.4. In order to have a better transfer gain, the transfer antenna is set as a directional antenna with an input power of 20 dBm (100 mW), and a half-wave dipole antenna is applied to the receiver.

In simulation scenarios, the receivers are positioned at intervals of 1.5 m at the height of 2 m, which means that the sampling interval is 1.5 m inside the tunnel.

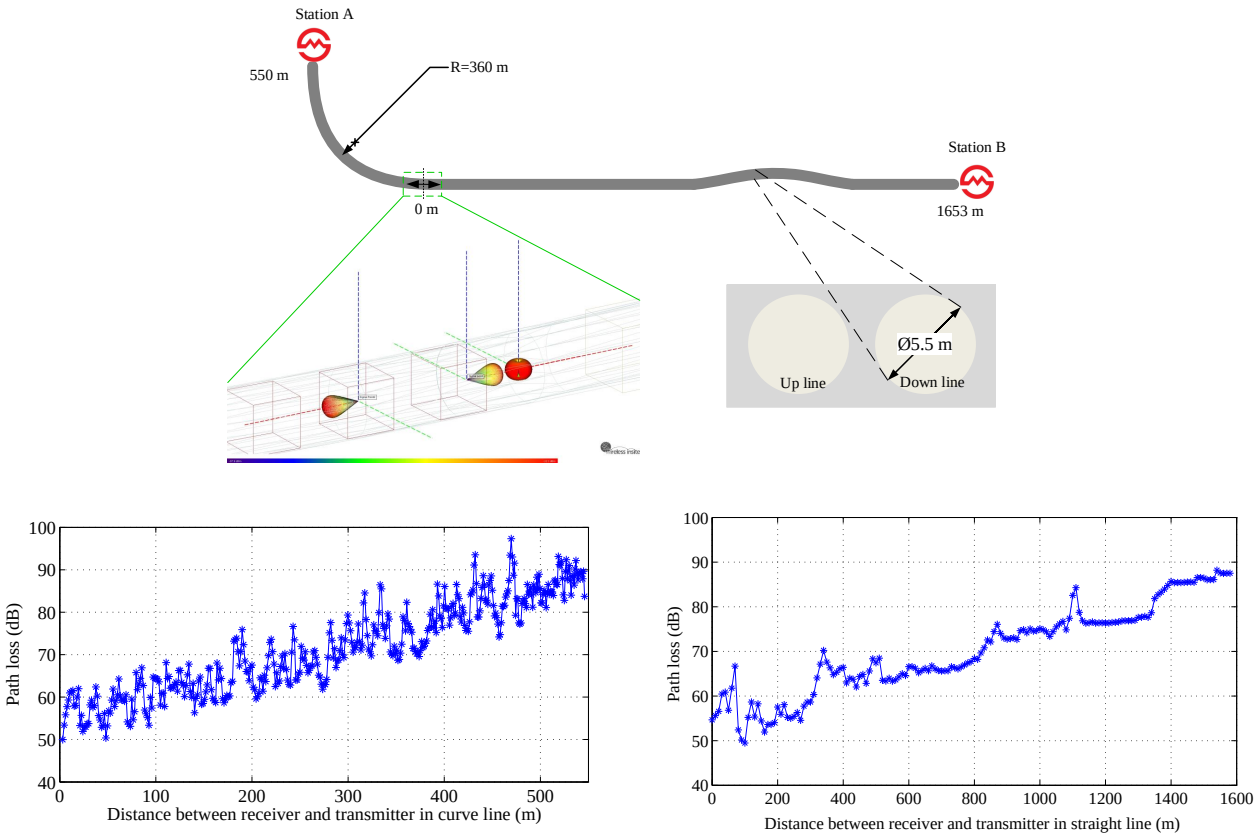


Figure 9.10: Simulation model and results

Fig. 9.10 reveals that the attenuation is lower than 100 dB in the curve tunnel line within 550 m. In the straight tunnel line, the path loss is not more than 90 dB when the distance is 1600 m. As indicated in the path loss simulation results, it is clear that the system can cover at least 1600 m in a straight line. In curve lines with a radius of 360 m, the signal can be received at a distance of more than 550 m.

9.5 Actual measurement

9.5.1 Measurement environment and configuration

The TTDMS prototype machine has the same measurement configuration as illustrated in Table 9.4. As shown in Fig. 9.11 (1), this prototype machine has three main parts: a

Table 9.4: Simulation parameters settings

Tunnel mode setting						
Tunnel size		Curve line		Tunnel material		
Φ(m)	L(m)	R(m)	L(m)	Material	Permittivity	Conductivity $(S.m^{-1})$
5.5	1653	360	550	Concrete	10	0.01
Carrier wave and antennas' setting						
Antenna Type	f (MHz)	Input power(dBm)	Antenna Type	Receiver threshold (dBm)	Gain (dBi)	Interval (m)
Directional	2400	20	Half-wave dipole	-100	6	1.5
Shooting-and-Bouncing Ray (SBR) setting						
Rays interval	number of reflections		Number of transmissions		Number of diffractions	
0.25	10		0		2	

host computer, a display monitor, and an antenna. The host computer implements the main calculation process and provides the input/output interfaces. The antenna is fixed at the top of the driver's room. The display monitor, which shows the distance and alarm information, is installed beside a Closed Circuit Television monitor. The application scenarios of TTDMS are shown in Fig. 9.11 (4) and (5).

Figure 9.11: Measurement: (1)TTDMS prototype machine, (2) Shanghai Metro line 7, (3) TTDMS antenna, (4) application scenario in tunnel, (5) application scenario on viaduct

The measurements were carried out in two scenarios. One scenario is in a static test line, and

another is in an actual operation line. The prototype machine outputs an initial distance result every 50 ms. The measurement data, which is a smoothing result of 10 initial results, is recorded every 0.5 seconds.

The static measurement is set so that the detected train was at a stop, while the following train is approaching or departing at a speed of 20 km/h. In this static measurement, the data recorded in one round trip measurement was obtained both inside the tunnel and on the viaduct lines.

In order to reflect the actual measurement situation, the actual distance measurements were implemented in Shanghai Metro line 7, China. As shown in Fig. 9.11 (2), the Metro line 7 is a west-southeast line of the Shanghai Metro network. The line's length is 44.35 km, it includes both underground and viaduct lines, and stops at 33 stations. The overview of the measurement data is shown in Table 9.5.

Table 9.5: Overview of the measurement data of Metro line 7

train-to-train distance	numbers of measurements
$>$ 800 m	453
600-800 m	289
300-600 m	123
$<$ 300 m	112
total	977

9.5.2 Measurement results

After these measurements, TTDMS measurement errors are obtained. The error bars represent the standard deviations of the mean values. As shown in Fig. 9.12, the data measured by TTDMS is comparable to the actual data, which is measured by the laser range-finder as the reference data. The measurement errors, which are the deviations between the measurement and actual data, are shown in Fig. 9.13. It is obvious that the scenarios show little differences in the location deviation for the different measurement environments. Additionally, one very interesting result is that the tunnel scenario has a lower measurement deviation than the viaduct. This is due to the tunnel's simple electromagnetic environment and waveguide transmission characteristics.

In the actual scenario, the data was collected in real operation metro trains. The data was recorded at different times and in different stations. The data from ATS was gathered as the reference system to draw a comparison as shown in Table 9.6. Given the data format

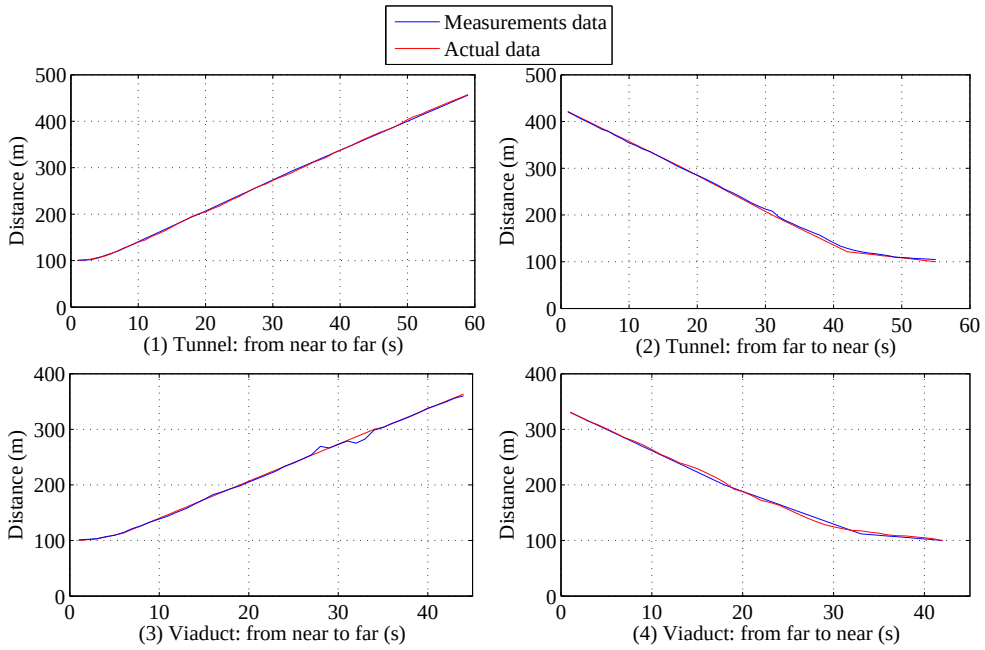


Figure 9.12: Static measurement

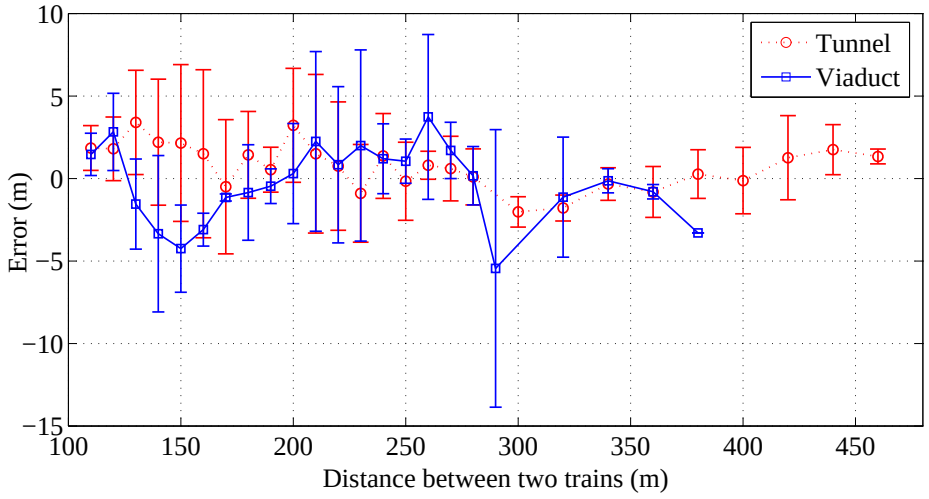


Figure 9.13: Static measurement errors

of ATS, only distances at time points are available. After the tests were performed and recorded, the deviation between the true and measured distance was calculated. Fig. 9.14 reveals that there is a steady increase in the measurement errors as the distance gets larger. The fitting line of the errors mean values is shown in equation 9.18.

$$Div = \alpha \times D + \beta \quad m \tag{9.18}$$

Table 9.6: Distance measurement in Metro line 7 (Partial)

Time	ID 1	ID 2	Stations / route	TTDMS (m)	ATS (m)
09:31:30-09:33:18	711	703	Dong'an Road	818-808	732-698
09:38:47-09:39:37	711	703	Changshu Road-Jing'an Temple	412-722	391-712
09:41:17-09:41:35	711	703	Jing'an Temple-Changping Road	554-486	552-469
09:42:02-09:42:09	711	703	Changping Road	591-654	554-598
09:43:20-09:43:23	711	703	Changping Road-Changshou Road	535-529	498-476
09:45:33-09:46:26	711	703	Changshou Road-Zhenping Road	426-260-413	424-258-412
09:48:34-09:50:03	711	703	Zhenping Road-Langao Road	504-700	451-620
09:50:08-09:50:32	711	703	Langao Road	646-537	570-480
09:50:39-09:50:51	711	703	Langao Road-Xincun Road	525-603	516-598
09:52:39-09:52:41	711	703	Xincun Road-Dahuasan Road	795-796	771-719
09:52:51-09:52:58	711	703	Xincun Road-Dahuasan Road	770-787	733-729
09:55:38-09:56:45	711	703	Dahuasan Road-Zhixing Road	1057-965	1006-933

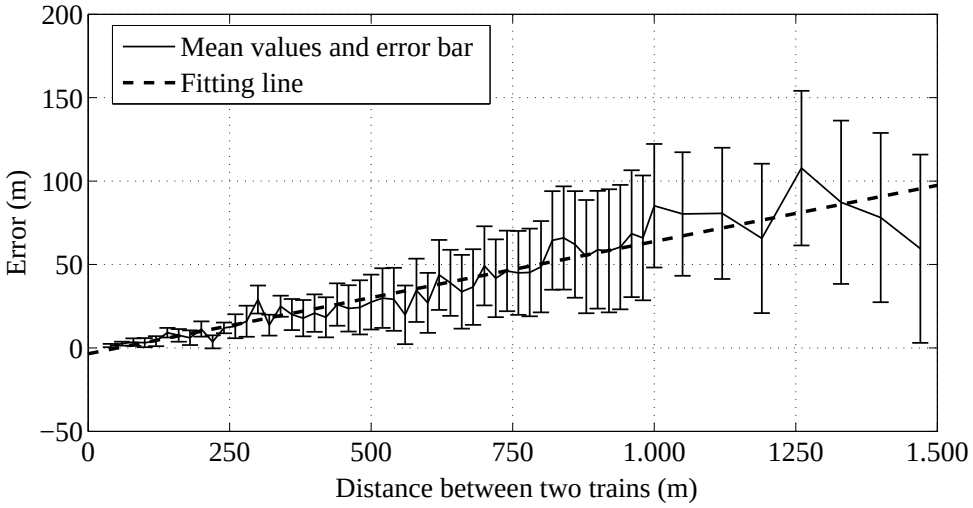


Figure 9.14: Measurement errors in Metro line 7

where, Div is the deviation between the true and measurement distance, α is the scope, $\alpha \in [0.06032, 0.07521]$ and $\beta \in [-2.935, 7.911]$. With the help of measurement inaccuracy results, an adjustment is available for future applications.

Possible reasons for measurement errors are the accuracy of the hardware, multi-path propagation and non-line-of-sight (NLOS). The hardware error is related to the efficiency and quality of the processor and chip. In this actual measurement, the deviations caused by NLOS is the main reason. Because of the reflection of the signal in tunnel space, the measurement distance differs from the actual distance and leads to a larger measured distance than the real one.

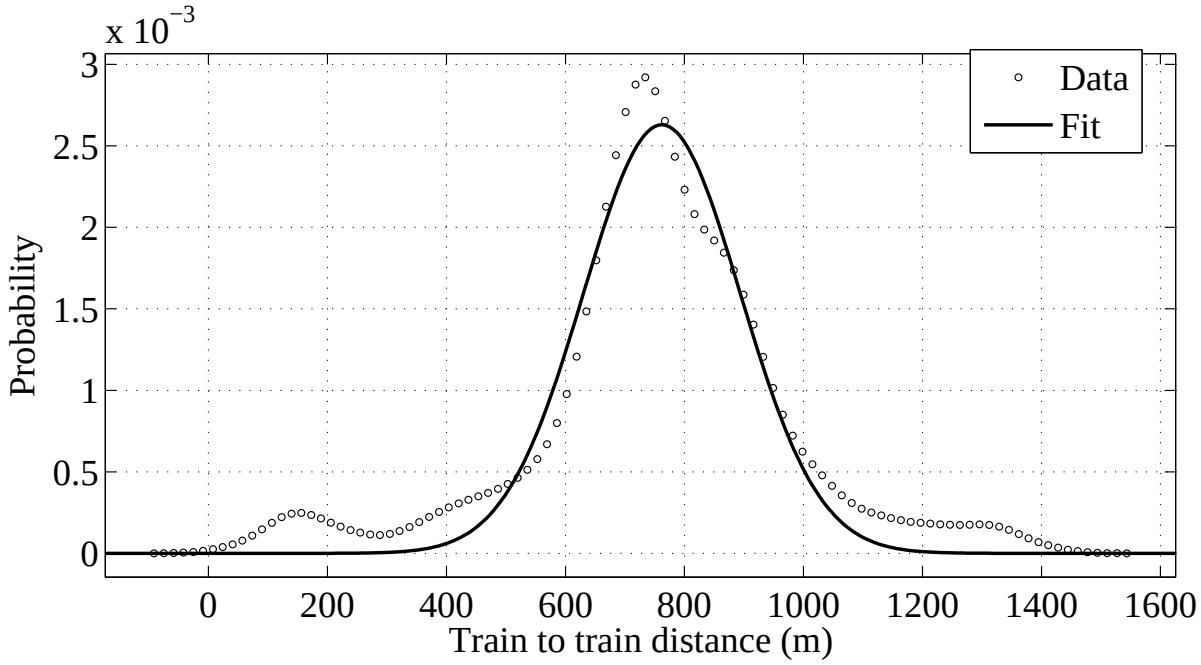


Figure 9.15: Probability distribution of train to train distance in operating (Metro line 7)

9.6 Summary

As the object of research, TTDMS can calculate the distance between two trains. We proposed the system architecture, application scenarios and block diagram. Furthermore, a prototype machine was assembled to obtain the actual measurement data. Numerical validations of TTDMS were implemented by applying mathematical calculation, software simulation and actual measurements. The results indicated that TTDMS was feasible in detecting distance and applicable in actual metro lines.

In this chapter, the amount of real measurement data is rather limited. Hence, the follow-up research works are foreseeable. To advance the system performance validation process, more real data should be collected. The benefit of getting a larger set of actual data is that the system's statistical characteristics become more apparent. Another research activity is to enhance the prototype machine. Based on the analysis data, the distance measurement results can be revised and its accuracy improved.

10 Practical implementation in the railway

As discussed in chapter 3, the movement authority presently applied in the Train Control System can only be generated by the Radio Block Center, and support one-dimensional distance information for the onboard equipment. The interlocking information and other trains' position are invisible to the individual train. Hence, if the onboard equipment outputs erroneous commands or abnormal communication between trains and Radio Block Center, train rear-end collisions are likely to happen. To satisfy the increasing safety demands of railway transportation especially in these abnormal scenarios,the MA+ is proposed in this thesis. Except for a wireless communication unit, the new MA+ does not require any other infrastructure.

With the help of train-centric communication technology, the MA+ can obtain the information of switches and trains within a certain scope. In this chapter, the MA+ detection range is estimated in section 10.1 based on a railway curve line model. Section 10.2 illustrates the MA+ implementation scenarios. Afterwards, the system logical model and working principles are introduced. In order to put the system into current ECTS-2, an application demo on the DMI is presented in section 10.3.

10.1 MA+ detection range estimation

A system has to be evaluated during its life cycle. The evaluation results provide essential information for the system design and improvement. The system performance represents the effectiveness of the system. The performance verification helps engineers have a better understanding of the system before its application in practice. Modeling and simulation can get the information about how a system reacts, without actually observing the data from

experiments in practice. With proper parameterization, the model can represent fundamental aspects of the system, and obtain data that are either difficult or expensive to replicate in the real world.

Any theoretical proposal should be advanced in engineering implementation, and then the proposal makes sense. Before the practice using, a suitable simulation based on an existing technology is essential. For the train-centric communication, different kinds of communication technologies are available, such as Global System for Mobile Communications (GSM), Code Division Multiple Access (CDMA), Wideband Code Division Multiple Access (WCDMA), and so on. In recent years, the evolution of the data communication technology promotes the application of wireless communication in rail transportations, for the Long-Term Evolution (LTE) as an example [109] [24]. Based on the high capacity and speed of LTE and less time delay (as shown in Table 10.1), direct communications among waysides and onboard equipment are available. It will enhance the safety level of the previous control method that relied principally on the information delivered from RBC. In this paper, the distance detection ability of the MA+ is discussed based on path loss.

Table 10.1: The maximum time delays comparison of mobile communication system

Communication system	Maximum time delay ($10^{-6}s$)	Maximum range error (m)
GSM	1.805	554
CDMA2000	0.813	344
WCDMA	0.130	39
LTE	0.030	9.7

The coverage of wireless signal limits the distance detection. Different factors have various influences on the signal quality. One tough transition environment is that in non-line-of-sight (NLOS) propagation with mountain barrier, for instance in railway curve lines, as shown in Fig. 10.1. Here, the path loss in curve line is treated as a single round obstacle for mathematical calculation.

where

h: height of the curve line above the straight line of the following train and detected train.

R: the radius of the curve line.

d_1, d_2: the tangent lines through the following train and detected train position.

The diffraction loss is influenced by the frequency, curve radius, and distance. The mathematical calculation is based on the ITU recommendation [110]. The diffraction loss A can

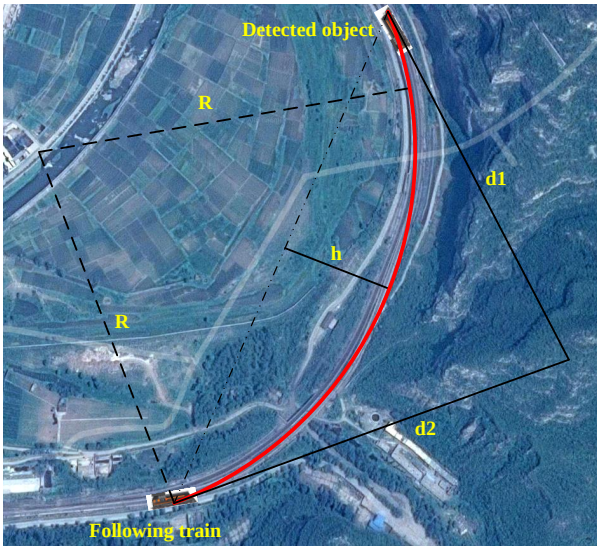


Figure 10.1: The power attenuation simulation in the curve line application scenario

be calculated by equation (10.1), $J(v)$ is the Fresnel-Kirchoff loss caused by equivalent blade shape barrier [110].

$$A = J(v) + T(m, n) \tag{10.1}$$

$$J\left(v\right) = -20\log\left(\frac{\sqrt{\left[1 - C\left(v\right) - S\left(v\right)\right]^2 + \left[C\left(v\right) - S\left(v\right)\right]^2}}{2}\right) \tag{10.2}$$

$C(v)$ and $S(v)$ are the real and imaginary parts of Fresnel integral as shown in equation (10.2), respectively. For the transition with a barrier blocking the line-of-sight transmission, the $J(v)$ can be approximately described by equation (10.4).

$$F_c\left(v\right) = \int_0^v \exp\left(j\frac{\pi s^2}{2}\right) ds = C\left(v\right) + jS\left(v\right) \tag{10.3}$$

$$J\left(v\right) = 6.9 + 20\log\left(\sqrt{\left(v - 0.1\right)^2 + 1} + v - 0.1\right) \tag{10.4}$$

where h and λ are in meters, λ is the wavelength, and d_1, d_2 are in kilometers.

$$v = 0.0316h\left[\frac{2\left(d_1+d_2\right)}{\lambda d_1 d_2}\right]^{1/2} \tag{10.5}$$

$T\left(m,n\right)$ is the additional loss caused by barrier curvature, which is the curve line radius R. When $mn \leq 4$, $T(m,n)$ equals equation (10.8), when $mn > 4$, $T(m,n)$ equals equation (10.9).

$$m = R\left[\frac{d_1+d_2}{d_1 d_2}\right] / \left[\frac{\pi R}{\lambda}\right]^{1/3} \tag{10.6}$$

$$n = h\left[\frac{\pi R}{\lambda}\right]^{2/3} / R \tag{10.7}$$

$$T\left(m,n\right) = 7.2m^{1/2} - \left(2-12.5n\right)m + 3.6m^{3/2} - 0.8m^2 \tag{10.8}$$

$$T\left(m,n\right) = -6 - 20\log\left(mn\right) + 7.2m^{1/2} - \left(2-17n\right)m + 3.6m^{3/2} - 0.8m^2 \tag{10.9}$$

In Europe, LTE frequencies are band 1/3/7/8/20. Band 8 is currently used mostly by GSM [111]. The band 8 is attractive from a coverage point of view due to the lower propagation losses. The band can be reused for LTE or HSPA. Band 8 and 20 hold the uplink frequencies 880-915 MHz and 832-862 MHz. The downlink frequencies are 925-960 MHz and 791-821 MHz [111].

The minimum railway curve radius is different in various railway lines. Several cases are shown in Table 10.2 [81].

Table 10.2: Comparison among different curve radiuses

Organization	JR	JR	DB	DB	SNC	SNCF	CRH
Item	Tokaido Shinkansen	Tokyo-Joetsu	Hannover-Würzburg	Köln-Rhein/Mann	Paris-Sud-Est	Atlantique	Beijing-Shanghai
Maximum design speed km/h	*null*	280	300	300	350	380	380
Maximum service speed km/h	300	275	250	*null*	270	300	300
Minimum curve radius m	4000	4000	7000	3350	4000	6250	7000

Fig. 10.2 shows the calculation results under the frequency of 930 MHz based on the diffraction loss. The simulation results indicate the interrelationship among the diffraction loss, detection distance, frequency, and radius. The received signal power should be greater than the sensitivity of a receiver, as shown in equation (10.10). The simulation results indicate that under the minimum curve radius 3350 m in Köln-Rhein/Mann line, the signal attenuation is 158.2 dB when the detection distance is 6100 m [94].

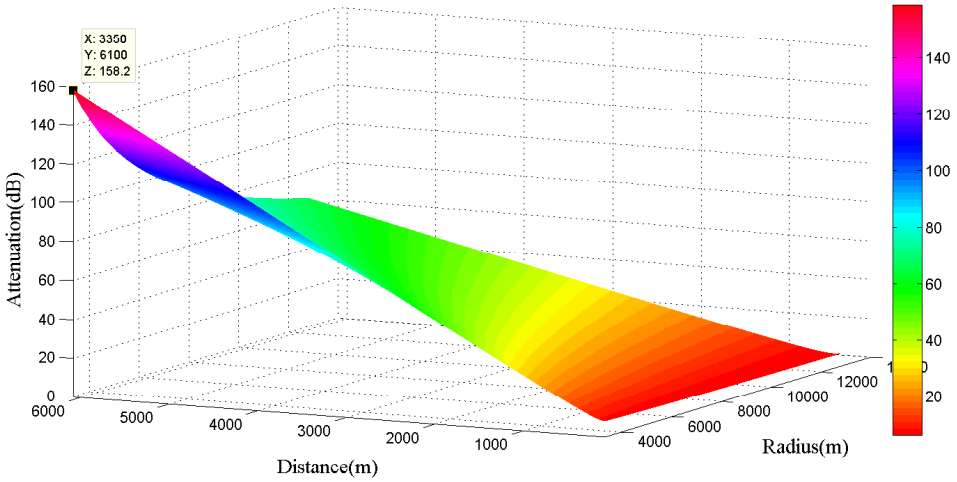


Figure 10.2: Wireless power attenuation with distance and radius

$$P_r\left(d\right)\left[dBm\right] = P_t\left[dBm\right] + G - A - PL_{\Delta} \tag{10.10}$$

$$G = 10\log\left(G_t G_r\right) \tag{10.11}$$

where $P_r\left(d\right)\left[dBm\right]$ is the receive power. $P_t\left[dBm\right]$ is the transfer power. G is the gain, G_t and G_r are the gain of the transfer and receiver antenna, respectively. PL_{Δ} is the attenuation caused by device and feeder cable.

For the received power, some experimental data are available to reference. In publications [112], the path loss measurements at the 930 MHz band in the different scenario were done. The empirical power of received signal models for suburban, open area, mountain area, and urban were proposed, as shown in equations 10.12, 10.13, 10.14, and 10.15, respectively. The

simulation result is shown in Fig. 10.3.

$$P_r(d) = 21.577 - 28.001 \log(d) \tag{10.12}$$

$$P_r(d) = 6.0246 - 21.2261 \log(d) \tag{10.13}$$

$$P_r(d) = 38.432 - 35.015 \log(d) \tag{10.14}$$

$$P_r(d) = 61.337 - 40.452 \log(d) \tag{10.15}$$

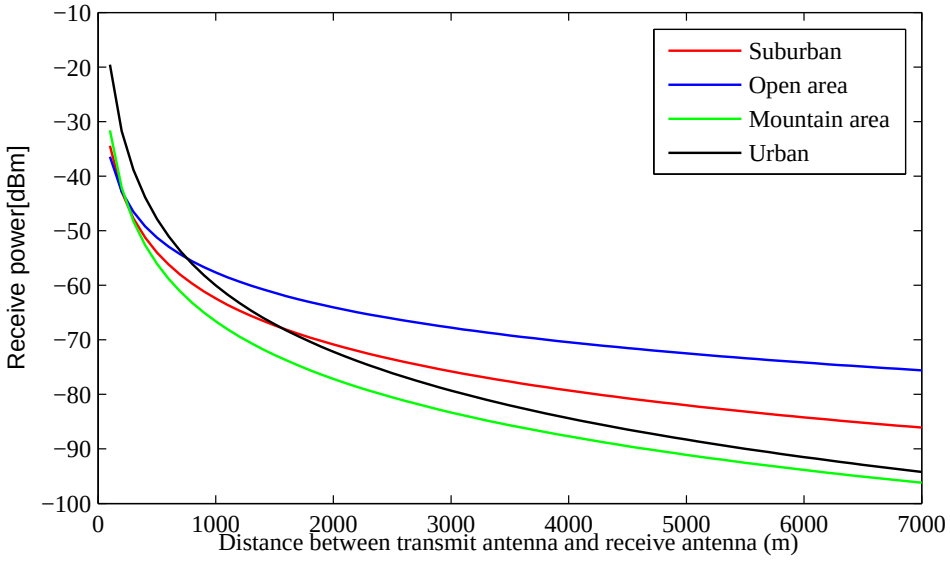


Figure 10.3: Receive signal power in different scenarios

In current practice application, the typical maximum path loss of LTE can be 163.5 dB [113]. In order to make the detection range as far as possible, many different methods can be used. For instance, choosing suitable wireless frequency spectrum according to the simulation result in Fig. 10.2; increasing transmitter power, enhancing receiver sensitivity, and building repeater stations. Hence, the detection range is not a limitation of the MA+ practical application, when compared with the train emergency braking distance (between 2300 and 2800 m depending on the actual speed, ICE-3, Germany).

Both the simulation result and practical results indicate that the train-centric communication is available at the technical level. The following part carries out the safety evaluation of the MA+.

10.2 MA+ implementation scenarios

A switch can lead a train on to a different path. Hence, obtaining the position and situation of the approaching switch is essential in the MA+ implementation. Train-to-switch communication provides fundamental vector information. The following steps are involved: surveillance, detect and appropriate avoidance, and output results.

- First, the following MA+ works in a monitoring mode. The switch announcement continuously broadcasts its location and position.
- Second, once the connection between the switch announcement and the following train-centric communication architecture establishes, the following system turns to the appropriate avoidance mode.
- Third, after obtaining the position and situation of the approaching switch, the system outputs results vary depending on different scenarios.

In the appropriate avoidance mode, the switch conditions are described as shown in Fig. 10.4. Each one switch unit connects two tracks. As shown, four different combinations of the switch conditions are discussed.

The red line represents the SMA. The blue rectangle indicates the normal switch position. The yellow one displays the reverse switch position. The switch name turns into green or yellow depending on the switch position. When the switch is in an uncertain position, it will be marked with red-dotted lines as shown in Fig. 10.4 (3) and (4).

It is important to note that, situations 3 and 4 do not exist in steady state when the interlocking system is working properly. ETCS is a critical safety system; the MA generation obeys specific fault-safety strategies. In order to reduce the frequency of false alarms, the following condition is considered. When the original MA is available and the database shows there is an approaching switch, but when the train-to-switch connection is not established, the system ignores the scenario, and no SMA is required. Among these four different scenarios, only one can trigger the SMA. As illustrated in Fig. 10.4 (3), under this particular scenario, switch split is likely to occur, along with derailments and side collisions.

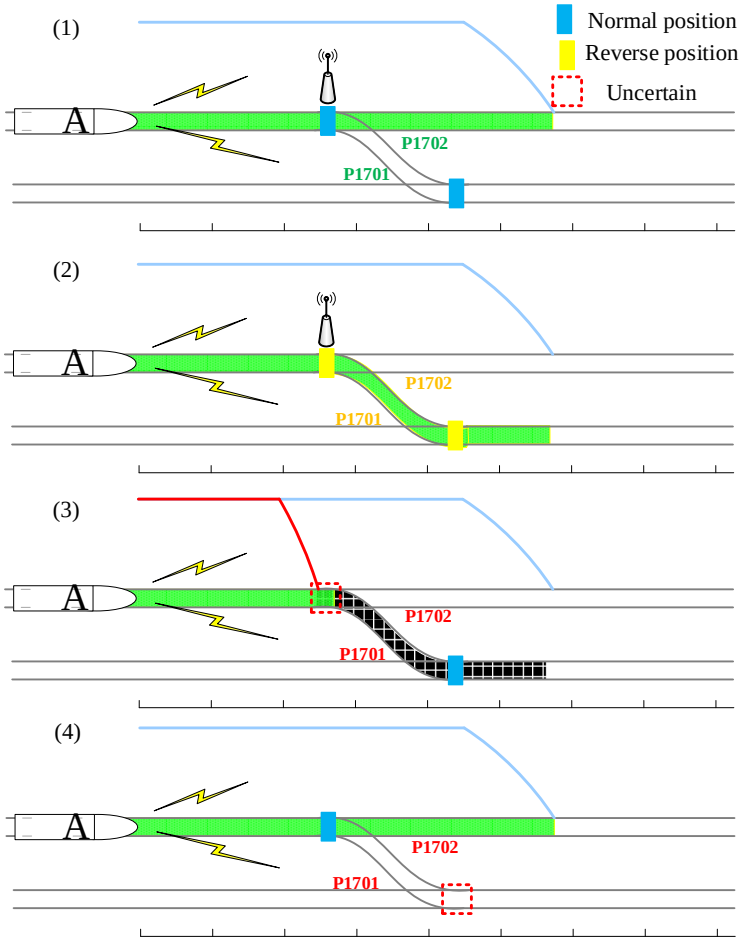


Figure 10.4: MA+ operates with approaching switch scenarios

If Train A detects another Train B in its detection range, Train A communicates with Train B, and gains the MA+ detail of Train B. Then, the scenarios can be divided into two main parts.

- There is no overlap of the two MA, as shown in Fig. 10.5 (1). The green part is the available extension area for the encountering a trains' MA. Under this scenario, the EOA of the MA can be reached without a risk for a hazardous situation[5].
- If the routes of Train A and Train B have an overlap, both of them have to activate the SMA to prevent collisions, as shown in Fig. 10.5 (2) and (3).

The practical scenarios are the combinations of approaching switches and encountering trains scenarios. When drivers are required to take responsibility, they can have an extended version of the surrounding switches and trains in a certain distance with the assistance of MA+.

[5]The train position calculated by D_LRBG is the train's head position, the absolute real length of the train should be considered when triggering SMA.

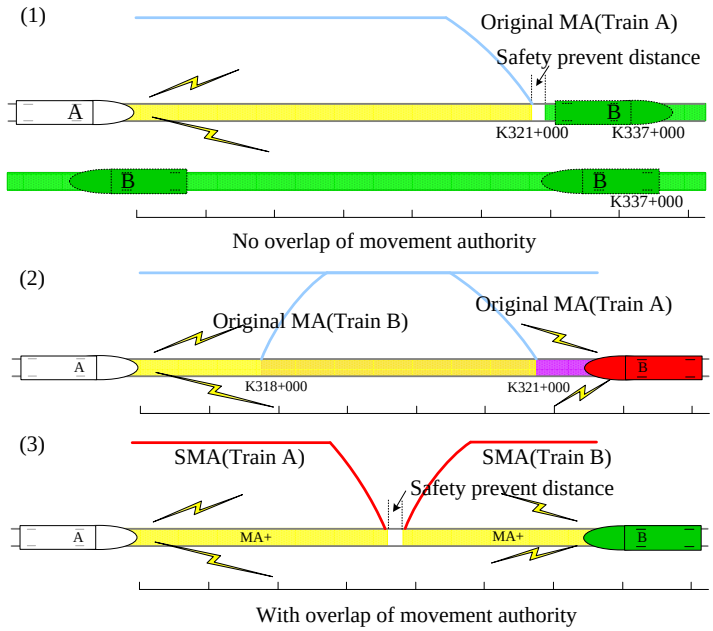


Figure 10.5: MA+ operates with encountering trains

For drivers, the benefits of MA+ implementation is that it helps them understand better the surrounding environment of tasks they have to perform, especially in special scenarios where the drivers have to make sure the situation of train ahead. For instance, the on-board equipment works in modes as ON SIGHT (OS), ISOLATION (IS), and so on. It is an efficient way to extend drivers' ability and improve safety.

The contemporary control system has a high safety level. However, if the former four defense layers result in failure, in this situation, the "Driver see and avoid" will be activated. The MA+ can output alarm and shorten the MA automatically, further more, this should be kept at Radio Silence if there is no potential accident, otherwise, it can also be turned on manually. Hence, it is clear that, the MA+ will not increase drivers' responsibility and workload. Additionally, it can also provide additional communication other than just the communication between trains and RBC.

10.3 An application demo on the DMI

The MA+ can be merged into the current train control system. In this section, a demo is proposed on the DMI of ETCS. The MA+ is combined with the planning information on DMI. Here we add MA+ information in orders and announcements of track conditions area

and these basic symbols as shown in Table 10.3.

Table 10.3: Symbol form/shape and descriptions

Symbol No.	Form/shape	Description
SW1		Switch in normal position
SW2		Switch in reverse position
SW3		Switch in reverse and normal position
SW4		Switch in normal and reverse position
TR1		Encountering train without collision risk
TR2		Encounter train with collision risk
TC1		Operation line
TC2		Empty line
TC3		Occupied line

Operation line and empty line are defined in the individual train view. Based on the switch conditions, different MA operation lines are optional. The driver will know which route is being implemented. As shown in Fig. 10.6, the L_{MA} is assumed to be 8000 m. It can be calculated from the variables of *L_ENDSECTION* and *L_SECTION(k)*, as shown in the equation 2.1. The train transfers from current operation line to empty line through SW2, which locates at 2800 m in front, as shown in Fig. 10.6 (2). If SW3 and SW4 are in hazardous conditions, trailed switch accidents may happen. SW3 will lead the following train to another track and switch split will occur. Hence, SMA is required and an alert is triggered. There is no risk of collisions to the following train under SW4, and no SMA action can be triggered automatically.

Once other trains were detected, and if there were no overlap between operation lines, the system will show the positions of other trains in a green symbol TR1. The angle shows the train operation direction. If there were overlaps between operation lines, SMA is immediately executed. Then, because of different encountering train directions, two different scenarios should be discussed. Scenario one, if the new EOA updates, there is still an overlap of the MA of two trains, both of two trains have to shorten their MA again based on their respective speed, distance, and location, as shown in Fig. 10.7 (1). Scenario two, if encountering trains and following trains have the same route direction, the new EOA will be updated depending on the end of the detected train, as shown in Fig. 10.7 (2).

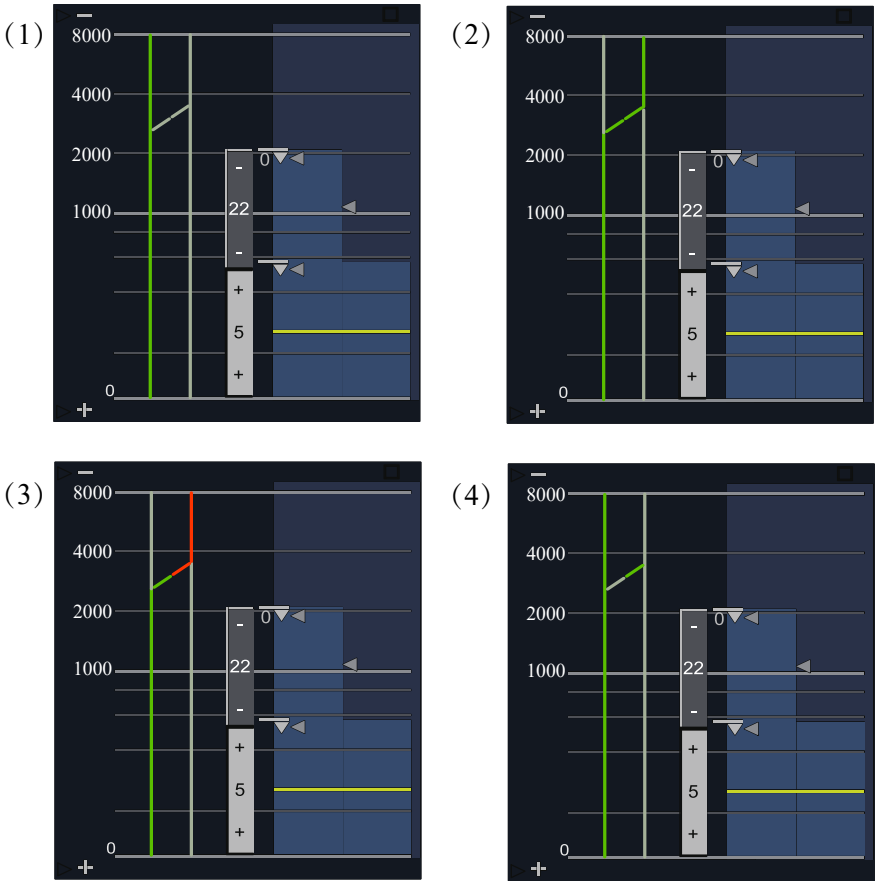


Figure 10.6: DMI operates with approaching switch scenarios

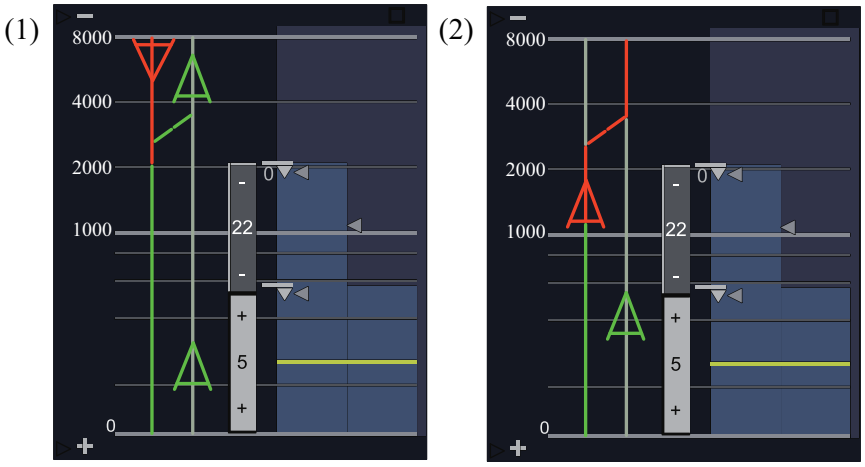


Figure 10.7: DMI with detected trains

10.4 Summary

In this chapter, the practical implementation of the MA+ in the railway was proposed. The MA+ detection range is estimated based on the mathematical calculation, which indicated that the LTE can be a possible solution in the railway application. Additionally, some actual measurements were done by other researchers also supported this conclusion. After discussed the MA+ implementation scenarios, an application demo on the DMI was proposed.

11 Conclusions and further work

In this chapter, first a summary of the methods proposed in this thesis is provided. Moreover, the approaches for the development and analysis of the system is discussed. Finally, the thesis is concluded by outlining the future work.

This thesis proposed an approach to assist the development and analysis of a distance measurement system. The system structure was presented and its correctness was validated by the CPN model. Based on the model-checking method, the functional safety validations that involve state and event were carried out. Further, a methodology for an encoding framework premised on an occurrence graph was proposed. Additionally, the encoding framework was applied to perform a static analysis of the execution time estimated in the latter part of this thesis.

Given that the functional analysis of the system is within the scope of states and functions validations, which both play important roles in this approach, this universal approach can also be applied in the development procedures of various systems.

11.1 Conclusions

System development is changing from informal text specifications and manual coding methods to a model-based and tool-supported system development and code automated generation process. In the model-based system development procedure, the formal method can be applied. Formal methods provide a means of developing a description of the system at some stage in the phases of requirements specification, design, and coding. A formal method generally offers a notation language, which permits deriving a description in that notation. Additionally, it is available to carry out the evaluation for different correctness properties by means of different forms. The modeling language is the foundation of the formal method.

In this thesis, Petri nets are selected as the modeling and verification language for the analysis of the train distance interval measurement system. For the development and validation of

the TTDMS, four steps and levels are considered. The thesis is organized as shown in the Fig. 11.1.

The first and second steps are about the conceptual and construction understanding of the system. These two steps are treated as the recognition level, which is carried out in the chapter 4. In this two steps, the system's requirements and system structure are analyzed, respectively. As the research target, the railway system safety is discussed in chapter 3. In this chapter, the essential of involving an additional train distance interval measurement system is proposed. The overall system safety improvement is calculated based on the reliability analysis. Additionally, the train collision model is introduced. With applying the train distance measurement system, some potential collisions can be avoided.

Steps	Levels	Chapters	Description
• Step 1	• Recognition level	• Chapter 3	• System requirements • System safety improvement
• Step 2		• Chapter 4	• System principle • System structure • Actual TTDMS reliability estimation • Performance improvement
• Step 3	• Modeling Level • Parameterization /simulation level	• Chapter 5 • Chapter 6 • Chapter 7 • Chapter 8	• Formal modeling process • Model validation and functional safety verification • Performance evaluation • Model based code generation and time estimation • Overlay system performance
• Step 4	• Physical level	• Chapter 9 • Chapter 10	• Metro • Railway

Figure 11.1: Thesis structure

Chapter 4 detailed the system principle of an enhanced movement authority system (MA+), which combines advantages of the train-centric communication with current movement authority mechanisms. The system structure, application scenarios, and algorithm flow chart of the MA+ are presented in the following. The MA+ provides two different methods of the train distance interval data acquisition: the normal one, which is based on the onboard equipment; the backup one, which is the main component of MA+ and based on the TOA distance estimation, is named as train to train distance measurement system (TTDMS). The performance improvement of the MA+ is evaluated by comparing the train safe distance interval protected by different solutions. The results indicate that the safe distance interval in MA+ is much less than it in ETCS-2 and ETCS-3 at any speed.

Step 3 is carried out in chapters 5 to 8. In chapter 5, the system evaluation methodology, formal methods for system evaluation and verification are discussed; the semantics and syntax of CPNs are expressed; the relationship between system structure and Petri nets are described; benefits of various Petri nets tool are compared; the system formalization and validation procedures, which are the modeling and parameterization/simulation levels, are implemented. During this step, the system is presented by using the formal method in the modeling level. In the parameterization and simulation level, the analysis of system functional safety and performance is carried out. After the model structure is validated, a universal approach is available to generate code frameworks from the occurrence graph.

With a new system involved, the overlay system performance is modified. As a collision fault tree model is introduced in chapter 3, the system dependability can be evaluated by applying CPNs. In chapter 8, a procedure is proposed to evaluate the fault tree by using CPNs, which combines event maintenance components, model correctness verification, time factors and mathematical calculation. This procedure breaks the limitations of many commercial tools, and it is flexible to be modified and free to use.

The last step is the combination of hardware and software, and this step is carried out on the physical level, which permits an actual system to implement both simulation, mathematical calculation, and actual testing processes. Hence, the practical implementations in the metro and railway system are introduced in chapters 9 and 10, respectively. In chapter 9, a speed based distance warning strategy is proposed for the metro system. The system availability is evaluated by applying stochastic Petri nets. Detection distance validation in both simulation and practical scenarios are considered. The results indicate that this system can carry out daily metro operations. For the implementation in the railway, the system detection range is estimated based on the mathematical calculation in chapter 10. Additionally, an application demo based on the DMI is proposed.

11.2 Outlook

For the further work, the deep combination of stochastic and Colored Petri nets should be taken into consideration. In other words, the stochastic analysis could be added to the existing model, which results in a various analysis based on the same model. Because the time parameters are treated differently in stochastic and Colored Petri nets, different models are required. The Colored stochastic Petri nets could be the possible solution. Here exists already some solutions, for example, REALIST (Universität Stuttgart), $\pi-$Tool (iVA), and so on. Additionally, the token on the place could be treated as stochastic, which can only

be modified through a transition. As a consequence, the system behavior can be more appropriately described and model could be further simplified.

For the distance distance measure system proposed in chapter 4, additional sensors and GNSS can be involved to increase the measurement accuracy. The data fusion technology will be taken into consideration, and the methodology introduced in this thesis is still available to do the relative analysis.

The discussion of the structural analysis in this thesis is based on the high-level Petri nets. The prerequisite is a full state space should be calculated, and no parameters are considered. Since the system with different token value is treated as a different state, which is limited by the software we applied, the parameterization is carried out after the structural analysis. The simulation in chapter 5 are implemented through Monte-Carlo-Algorithms testing, which normally associated with a confidence interval. Hence, the results may not be the same with mathematical calculations. This could be solved when the deep combination of stochastic and Colored Petri nets is available. At that time, the Markovian Analysis could provide more precise results. The model based code generation and time estimation are introduced principally, and an actual procedure will be the further work, which can implement the transition from safe model to safe code.

More examples should be involved in further research work. In doing so, a more organized and constructive validation procedure will be developed to evaluate and validate the system structure. Although the approach does have its limitations, a more efficient CPN-based development procedure can be expected.

For the overlay system performance evaluation in chapter 3, only fault tree analysis is applied. Since the causal relationship between events that lead to train collisions is represented by a fault tree, the limitation is that the model could be different from the realistic model. What is more, the parameterization procedure is limited by the data resources.

Due to business and policy restrictions, the practical prototype machine is implemented in the metro only. Hence, there is no actual measurement data in the railway application. In chapter 10, the application demo on the DMI is a hypothesis. Since the current DMI has been developed with a view to optimizing information provision and providing drivers with the information they need, and no more than that. Any changes to the DMI and information provided would, therefore, need careful consideration. Additionally, the MA+ will provide drivers with additional information, which may increase the drivers' workload. These issues can be figured out by consulting the train drivers for more information.

Overall, however, the approach proposed in this thesis holds great promise for the development and verification of the train-to-train distance measurement system, which will enhance the safety of metro transportation systems as well as railway systems.

Bibliography

[1] E. Schnieder, L. Schnieder, and J. Mueller, "Conceptual foundation of dependable systems modelling," in *Dependable Control of Discrete Systems*, vol. 2, no. 1, 2009, pp. 198–202.

[2] H. Dong, B. Ning, B. Cai, and Z. Hou, "Automatic train control system development and simulation for high-speed railways," *IEEE circuits and systems magazine*, vol. 10, no. 2, pp. 6–18, 2010.

[3] "Ertms/etcs - baseline 3, 2008. system requirements specification. chapter 3, principles. subset-026-3, issue 3.0.0. 23," December 2008.

[4] X. Cheng, L. Yang, and X. Shen, "D2d for intelligent transportation systems: A feasibility study," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 1784–1793, 2015.

[5] D. Chen, R. Chen, Y. Li, and T. Tang, "Online learning algorithms for train automatic stop control using precise location data of balises," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1526–1535, 2013.

[6] D. Lu, "Gnss for train localisation performance evaluation and verification," Ph.D. dissertation, Technische Universität Braunschweig, 2014.

[7] D. Chen, Y.-S. Fu, B. Cai, and Y.-X. Yuan, "Modeling and algorithms of gps data reduction for the qinghai–tibet railway," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 753–758, 2010.

[8] G. De Angelis, G. Baruffa, and S. Cacopardi, "Gnss/cellular hybrid positioning system for mobile users in urban scenarios," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 313–321, 2013.

[9] P. Papadimitratos, A. De La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 84–95, 2009.

[10] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1067–1080, 2007.

[11] Z. Farid, R. Nordin, and M. Ismail, "Recent advances in wireless indoor localization techniques and system," *Journal of Computer Networks and Communications*, vol. 2013, 2013.

[12] M. L. Bencheikh and Y. Wang, "Joint dod-doa estimation using combined esprit-music approach in mimo radar," *Electronics Letters*, vol. 46, no. 15, pp. 1081–1083, 2010.

[13] S. Semmelrodt and R. Kattenbach, "Investigation of different fading forecast schemes for flat fading radio channels," in *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th*, vol. 1. IEEE, 2003, pp. 149–153.

[14] M. Hartong, R. Goel, and D. Wijesekera, "Positive train control (ptc) failure modes," *Journal of King Saud University - Science*, vol. 23, no. 3, pp. 311–321, 2011.

[15] J. Wang, J. Wang, C. Roberts, L. Chen, and Y. Zhang, "A novel train control approach to avoid rear-end collision based on geese migration principle," *Safety science*, vol. 91, pp. 373–380, 2017.

[16] N. M. Zeigler, "Positive train control: safety, effectiveness, and security," Ph.D. dissertation, Utica College, 2016.

[17] S. Dirk, H. R. Sebastian, J. Welte, and E. Schnieder, "Integration of petri nets into stamp/cast on the example of wenzhou 7.23 accident," *IFAC Proceedings Volumes*, vol. 46, no. 25, pp. 65–70, 2013.

[18] H. Dong, B. Ning, Y. Chen, X. Sun, D. Wen, Y. Hu, and R. Ouyang, "Emergency management of urban rail transportation based on parallel systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 627–636, 2013.

[19] J. Wang, J. Wang, C. Roberts, and L. Chen, "Parallel monitoring for the next generation of train control systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 330–338, 2015.

[20] A. Lacher, D. Maroney, and A. Zeitlin, "Unmanned aircraft collision avoidance–technology assessment and evaluation methods (2008)," *The MITRE Corporation: McLean, VA*.

[21] T. Albrecht, K. Luddecke, and J. Zimmermann, "A precise and reliable train positioning system and its use for automation of train operation," in *2013 IEEE International Conference on Intelligent Rail Transportation Proceedings*. IEEE, aug 2013.

[22] J. Goikoetxea, "Roadmap towards the wireless virtual coupling of trains," in *International Workshop on Communication Technologies for Vehicles*. Springer, 2016, pp. 3–9.

[23] T. Schumann, "Increase of capacity on the shinkansen high-speed line using virtual coupling," *International Journal of Transport Development and Integration*, vol. 1, no. 4, pp. 666–676, 2017.

[24] B. Ai, X. Cheng, T. Kurner, Z.-D. Zhong, K. Guan, R.-S. He, L. Xiong, D. W. Matolak, D. G. Michelson, and C. Briso-Rodriguez, "Challenges toward wireless communications for high-speed railway," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2143–2158, 2014.

[25] J. Moreno, J. M. Riera, L. De Haro, and C. Rodriguez, "A survey on future railway radio communications services: challenges and opportunities," *Communications Magazine, IEEE*, vol. 53, no. 10, pp. 62–68, 2015.

[26] A. Lehner, T. Strang, and C. R. García, "A reliable surveillance strategy for an autonomous rail collision avoidance system," in *Proceedings of the 15th ITS World Congress, New York, USA*, 2008.

[27] E. Schnieder, M. Chouikha, S. Einer, and M. M. Zu Hörste, "Basysnet–an integrated approach for automated control system development," in *Petri Net Technology for Communication-Based Systems*. Springer, 2003, pp. 352–362.

[28] D. Meedeniya and I. Perera, "Model based software design: Tool support for scripting in immersive environments," in *Industrial and Information Systems (ICIIS), 2013 8th IEEE International Conference on*. IEEE, 2013, pp. 248–253.

[29] R. Wang and C. H. Dagli, "An executable system architecture approach to discrete events system modeling using sysml in conjunction with colored petri net," in *Systems Conference, 2008 2nd Annual IEEE*. IEEE, 2008, pp. 1–8.

[30] P. Diekhake, "Systematische modellierung und analyse verteilter automatisierungssysteme," Ph.D. dissertation, Technische Universität Carolo-Wilhelmina zu Braunschweig, 2016.

[31] A. Zimmermann and G. Hommel, "A train control system case study in model-based real time system design," in *Parallel and Distributed Processing Symposium, 2003. Proceedings. International*. IEEE, 2003, pp. 8–pp.

[32] L. Jansen, M. M. Zu Horste, and E. Schnieder, "Technical issues in modelling the european train control system (etcs) using coloured petri nets and the design/CPN tools," 1998.

[33] F. Netjasov, A. Vidosavljevic, V. Tosic, M. H. Everdij, and H. A. Blom, "Development, validation and application of stochastically and dynamically coloured petri net model of acas operations for safety assessment purposes," *Transportation Research Part C: Emerging Technologies*, vol. 33, pp. 167–195, 2013.

[34] H. Song, J. Liu, and E. Schnieder, "Validation, verification and evaluation of a train to train distance measurement system by means of colored petri nets," *Reliability Engineering & System Safety*, vol. 164, pp. 10–23, 2017.

[35] D. Tokody, I. J. Mezei, and G. Schuster, "An overview of autonomous intelligent vehicle systems," in *Vehicle and Automotive Engineering*. Springer, 2017, pp. 287–307.

[36] I. David and R. Rituraj, "Design of automated unmanned railway level crossing system using wheel detector (sensor) technology."

[37] E. Schnider, "Traffic safety and availability - contradiction or attraction," in *Proceedings of the 2nd International Symposium of Networks for Mobility*, 2004.

[38] Y. Baba, Y. Tateishi, K. Mori, S. Aoyagi, K. Takeshi, S. Saito, Y. Suzuki, and T. Watanabe, "Radio train control system (atacs)," *JR East Technical Review*, no. 3, 2004.

[39] E. Schnieder, L. Schnieder, and J. R. Mueller, "Conceptual foundation of dependable systems modelling," in *Dependable Control of Discrete Systems*, ser. Dependable Control of Discrete Systems, vol. 2, 2009, pp. 198–202, 1.

[40] T. Strang, M. Meyer zu Hörste, and X. Gu, "A railway collision avoidance system exploiting ad-hoc inter-vehicle communications and galileo," in *Proceedings*, 2006.

[41] W. M. Goble, *Control systems safety evaluation and reliability.* ISA, 2010.

[42] "Uic safety database-report 2014 significant accident 2013," International Union of Railways, Tech. Rep., 2014.

[43] A. Nash, D. Huerlimann, J. Schütte, and V. P. Krauss, "Railml+ a standard data interface for railroad applications," *WIT Transactions on The Built Environment*, vol. 74, 2004.

[44] Y. Zhao, "Mobile phone location determination and its impact on intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 1, pp. 55–64, 2000.

[45] R. J. Kephart and M. S. Braasch, "See-and-avoid comparison of performance in manned and remotely piloted aircraft," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 25, no. 5, pp. 36–42, 2010.

[46] X. Mao, D. Inoue, S. Kato, and M. Kagami, "Amplitude-modulated laser radar for range and speed measurement in car applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 408–413, 2012.

[47] T. Strang and A. Lehner, "On the applicability of tetra for direct train-to-train beaconing," 2013.

[48] J. K. Baker, "Advanced automatic train control pioneered in san francisco," *Railway Gazette International*, vol. 158, no. 6, 2002.

[49] F. Perich, E. Morgan, O. Ritterbush, M. McHenry, and S. D'Itri, "Efficient dynamic spectrum access implementation," in *MILITARY COMMUNICATIONS CONFERENCE, 2010-MILCOM 2010.* IEEE, 2010, pp. 1887–1892.

[50] Y. Elhillali, C. Tatkeu, A. Rivenq, J.-M. Rouvaen, and J.-P. Ghys, "Location and communication using cooperative radar system dedicated to guided transports," *Transportation Research Part C: Emerging Technologies*, vol. 16, no. 2, pp. 141–152, 2008.

[51] J. S. Kwak and J. H. Lee, "Infrared transmission for intervehicle ranging and vehicle-to-roadside communication systems using spread-spectrum technique," *IEEE Transactions on intelligent transportation systems*, vol. 5, no. 1, pp. 12–19, 2004.

[52] M. S. Braasch and A. Van Dierendonck, "GPS receiver architectures and measurements," *Proceedings of the IEEE*, vol. 87, no. 1, pp. 48–64, 1999.

[53] L. Ding, F. Geng, and J. Chen, "Radar principle," *Xian: Electronic technology*, 2002.

[54] S. Midya and R. Thottappillil, "An overview of electromagnetic compatibility challenges in european rail traffic management system," *Transportation Research Part C: Emerging Technologies*, vol. 16, no. 5, pp. 515–534, 2008.

[55] J. Wang, "CTCS-2I: New train control system suitable for trains with speeds up to 350 km/h," *Journal of Transportation Engineering*, vol. 137, no. 5, pp. 327–332, 2010.

[56] E. CENELEC, "50128-railway applications-communication, signalling and processing systems-software for railway control and protection systems," *Book EN*, vol. 50128, 2012.

[57] M. Baehr, "Evaluation methodology," *Elmhurst College*, p. 4, 2004.

[58] C. Lijie, T. Tao, Z. Xianqiong, and E. Schnieder, "Verification of the safety communication protocol in train control system using colored petri net," *Reliability Engineering & system safety*, vol. 100, pp. 8–18, 2012.

[59] N. Khakzad, F. Khan, and P. Amyotte, "Safety analysis in process facilities: Comparison of fault tree and bayesian network approaches," *Reliability Engineering & System Safety*, vol. 96, no. 8, pp. 925–932, 2011.

[60] L. Zhang, X. Wu, M. J. Skibniewski, J. Zhong, and Y. Lu, "Bayesian-network-based safety risk analysis in construction projects," *Reliability Engineering & System Safety*, vol. 131, pp. 29–39, 2014.

[61] A. Lisnianski, D. Elmakias, D. Laredo, and H. B. Haim, "A multi-state markov model for a short-term reliability analysis of a power generating unit," *Reliability Engineering & System Safety*, vol. 98, no. 1, pp. 1–6, 2012.

[62] T. K. Nguyen, J. Beugin, and J. Marais, "Method for evaluating an extended fault tree to analyse the dependability of complex systems: Application to a satellite-based railway system," *Reliability Engineering & System Safety*, vol. 133, pp. 300–313, 2015.

[63] D. Wu, "Verifiable design of a satellite-based train control system with petri nets," Ph.D. dissertation, Technische Universität Braunschweig, 2014.

[64] A. Kleyner and V. Volovoi, "Application of petri nets to reliability prediction of occupant safety systems with partial detection and repair," *Reliability Engineering & System Safety*, vol. 95, no. 6, pp. 606–613, 2010.

[65] R. Cheng, J. Zhou, D. Chen, and Y. Song, "Model-based verification method for solving the parameter uncertainty in the train control system," *Reliability Engineering & System Safety*, vol. 145, pp. 169–182, 2016.

[66] M. zu Hörste, H. Hungar, and E. Schnieder, "Modelling functionality of train control systems using petri nets," in *Towards a Formal Methods Body of Knowledge for Railway Control and Safety Systems, 2013 FM-RAIL-BOK Workshop*, 2013, pp. 46–50.

[67] M. Horste and E. Schnieder, "Modeling train control systems with petri nets-a functional reference-architecture," in *Systems, Man, and Cybernetics, 2000 IEEE International Conference on*, vol. 4. IEEE, 2000, pp. 3081–3086.

[68] J. Tang, M. A. Piera, and T. Guasch, "Coloured petri net-based traffic collision avoidance system encounter model for the analysis of potential induced collisions," *Transportation Research Part C: Emerging Technologies*, vol. 67, pp. 357–377, 2016.

[69] K. Jensen and L. M. Kristensen, *Coloured Petri nets: modelling and validation of concurrent systems*. Springer Science & Business Media, 2009.

[70] K. Jensen, *Coloured Petri nets: basic concepts, analysis methods and practical use*. Springer Science & Business Media, 2013, vol. 1.

[71] S. Christensen and K. H. Mortensen, "Design/CPN ASK-CTL manual," *University of Aarhus*, 1996.

[72] K. Jensen, S. Christensen, and L. M. Kristensen, "CPN tools state space manual," *Department of Computer Science, Univerisity of Aarhus*, 2006.

[73] E. Schnieder, *Methoden der Automatisierung: Beschreibungsmittel, Modellkonzepte und Werkzeuge für Automatisierungssysteme*. Springer-Verlag, 1999.

[74] L. Schnieder, "Formalisierte Terminologien technischer Systeme und ihrer Zuverlässigkeit," Ph.D. dissertation, DLR-Institut für Verkehrssystemtechnik, 2010.

[75] E. Schnieder and L. Schnieder, "Terminologische Präzisierung des Systembegriffs," *atp edition*, vol. 52, no. 09, pp. 46–59, 2010.

[76] A. Zimmermann and G. Hommel, "Towards modeling and evaluation of etcs real-time communication and operation," *Journal of Systems and Software*, vol. 77, no. 1, pp. 47–54, 2005.

[77] S. Distefano and A. Puliafito, "Reliability and availability analysis of dependent-dynamic systems with drbds," *Reliability Engineering & System Safety*, vol. 94, no. 9, pp. 1381–1393, 2009.

[78] I. E. Commission *et al.*, "IEC 61508: Functional safety of electrical," *Electronic/Programmable Electronic Safety-Related Systems*, 1999.

[79] K. Jensen and L. M. Kristensen, "Colored petri nets: modelling and validation of concurrent systems," *Springer Verlag, ISBN*, vol. 978, no. 3, p. 642, 2009.

[80] A. V. Ratzer, L. Wells, H. M. Lassen, M. Laursen, J. F. Qvortrup, M. S. Stissing, M. Westergaard, S. Christensen, and K. Jensen, "Cpn tools for editing, simulating, and analysing coloured petri nets," in *International Conference on Application and Theory of Petri Nets*. Springer, 2003, pp. 450–462.

[81] M. Lindahl, "Track geometry for high-speed railways," TRITA-FKT Report, Tech. Rep., 54.

[82] M. Akkouchi, "On the convolution of exponential distributions," *J. Chungcheong Math. Soc*, vol. 21, no. 4, pp. 501–510, 2008.

[83] H. Song and E. Schnieder, "Development and validation of a distance measurement system in metro lines," *IEEE Transactions on Intelligent Transportation Systems*, 2018.

[84] S. Philippi, "Automatic code generation from high-level petri-nets for model driven systems engineering," *Journal of Systems and Software*, vol. 79, no. 10, pp. 1444–1455, 2006.

[85] S. Parthasarthy and E. Schnieder, "The explication problem: the achilles heel of formal methods," *Entwicklung und Betrieb komplexer Automatisierungssysteme, EKA*, vol. 99, pp. 93–103, 1999.

[86] J. Hansen, S. A. Hissam, and G. A. Moreno, "Statistical-based WCET estimation and validation," in *Proceedings of the 9th Intl. Workshop on Worst-Case Execution Time (WCET) Analysis*, 2009.

[87] S. M. Petters, "How much worst case is needed in wcet estimation," in *2nd International Workshop on Worst Case Execution Time Analysis*, 2002, pp. 111–112.

[88] S. Nadarajah, "Exact distribution of the linear combination of p gumbel random variables," *International Journal of Computer Mathematics*, vol. 85, no. 9, pp. 1355–1362, 2008.

[89] J. Magott and P. Skrobanek, "Timing analysis of safety properties using fault trees with time dependencies and timed state-charts," *Reliability Engineering & System Safety*, vol. 97, no. 1, pp. 14–26, 2012.

[90] K. Buchacker, "Modeling with extended fault trees," in *High Assurance Systems Engineering, 2000, Fifth IEEE International Symposim on. HASE 2000.* IEEE, 2000, pp. 238–246.

[91] A. Lindhe, T. Norberg, and L. Rosen, "Approximate dynamic fault tree calculations for modelling water supply risks," *Reliability Engineering & System Safety*, vol. 106, pp. 61–71, 2012.

[92] G. K. Palshikar, "Temporal fault trees," *Information and Software Technology*, vol. 44, no. 3, pp. 137–150, 2002.

[93] M. Malhotra and K. S. Trivedi, "Dependability modeling using petri-nets," *IEEE Transactions on Reliability*, vol. 44, no. 3, pp. 428–440, 1995.

[94] H. Song and E. Schnieder, "Modeling of railway system maintenance and availability by means of colored petri nets," *EKSPLOATACJA I NIEZAWODNOSC- Maintenance And Reliability*, vol. 20, no. 2, pp. 232–239, 2018.

[95] B. Bertsche, *Reliability in automotive and mechanical engineering: determination of component and system reliability.* Springer Science & Business Media, 2008.

[96] P. E. Miyagi and L. Riascos, "Modeling and analysis of fault-tolerant systems for machining operations based on petri nets," *Control Engineering Practice*, vol. 14, no. 4, pp. 397–408, 2006.

[97] M. Handbook, "Mil-hdbk-338b," *US Department of Defense*, vol. 1, 1998.

[98] L. Wells, "Performance analysis using cpn tools," in *Proceedings of the 1st international conference on Performance evaluation methodolgies and tools*. ACM, 2006, p. 59.

[99] J. A. B. Geymayr and N. F. F. Ebecken, "Fault-tree analysis: a knowledge-engineering approach," *IEEE Transactions on Reliability*, vol. 44, no. 1, pp. 37–45, 1995.

[100] K. C. Kapur and M. Pecht, *Reliability engineering*. John Wiley & Sons, 2014.

[101] W. J. Davis, *The tractive resistance of electric locomotives and cars*. General Electric, 1926.

[102] H. Song, T. Shen, and W. Wang, "Train-centric communication-based close proximity driving train movement authority system," *IEEE Intelligent Transportation Systems Magazine*, pp. 1–1, 2018.

[103] E. FFFIS, "class 1 requirements," 2000.

[104] L. M. Quiroga, U. Becker, and E. Schnieder, "Das Petrinetz Modellierungs-und-Analysetool π-Tool," *at-Automatisierungstechnik*, vol. 62, no. 6, pp. 436–445, 2014.

[105] Y. Zhao, R. Adve, and T. J. Lim, "Symbol error rate of selection amplify-and-forward relay systems," *IEEE Communications Letters*, vol. 10, no. 11, 2006.

[106] B. EN, "13452-1: 2003," *Railway applications–braking–mass transit brake systems*.

[107] A. Steimel, *Electric traction-motive power and energy supply: basics and practical experience*. Oldenbourg Industrieverlag, 2008.

[108] O. Shoewu and A. Adedipe, "Investigation of radio waves propagation models in nigerian rural and sub-urban areas," *American Journal of Scientific and Industrial Research*, vol. 1, pp. 227–232, 2010.

[109] L. Lei, J. Lu, Y. Jiang, X. S. Shen, Y. Li, Z. Zhong, and C. Lin, "Stochastic delay analysis for train control services in next-generation high-speed railway communications system," 2016.

[110] *ITU-R Recommendation P.526-11, "Propagation by diffraction"*, 2009.

[111] H. Holma and A. Toskala, *LTE for UMTS: Evolution to LTE-advanced*. John Wiley & Sons, 2011.

[112] R. He, B. Ai, Z. Zhong, A. F. Molisch, R. Chen, and Y. Yang, "A measurement-based stochastic model for high-speed railway channels," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1120–1135, 2015.

[113] H. Abid, T. C. Chung, S. Lee, and S. Qaisar, "Performance analysis of LTE smartphones-based vehicle-to-infrastrcuture communication," in *2012 9th International Conference on Ubiquitous Intelligence and Computing and 9th International Conference on Autonomic and Trusted Computing*. IEEE, sep 2012.